



Escola Politècnica Superior
d'Enginyeria de Vilanova i la Geltrú

UNIVERSITAT POLITÈCNICA DE CATALUNYA

TREBALL FINAL DE GRAU

TÍTOL: DISPOSITIU AUTOMÀTIC PER JUGAR A ESCACS

AUTOR: CIVILL CORTADA, MARÇAL

DATA DE PRESENTACIÓ: 24 / 05 / 2018

COGNOMS: CIVILL CORTADA

NOM: MARÇAL

TITULACIÓ: GRAU EN ENGINYERIA MECÀNICA

PLA: GRAU

DIRECTOR: SOLÉ ROVIRA, JOAN

DEPARTAMENT: DEPARTAMENT D'ENGINYERIA MECÀNICA

Aquest Projecte té en compte aspectes mediambientals: ☐ Sí ☒ No

RESUM

En aquest document s'exposen el plantejament, procediments i resultats d'un projecte que té per objectiu dissenyar i fabricar un sistema mecatrònic capaç de jugar a escacs contra una persona en un taulell físic. El sistema és capaç d'analitzar la situació del joc a partir d'una imatge, a partir d'aquesta informació calcular les seves pròpies jugades i finalment moure les seves pròpies peces.

El programa informàtic està escrit en Python completament. El bloc principal del programa està dissenyat per captar la informació del taulell a partir d'una foto capturada amb una càmera, analitzar la imatge i extreure'n la informació necessària per conèixer la situació del joc. Per aquest procés s'han utilitzat diverses funcions de les llibreries OpenCV, llibreria de codi obert per a l'anàlisi d'imatge i NumPy, llibreria de codi obert per a l'ús de matrius.

Amb la informació extreta de l'anàlisi d'imatge és capaç de calcular les jugades, per a aquest procés s'han utilitzat funcions de la llibreria Python-Chess per a generar i actualitzar el taulell virtual i el motor d'escacs Stockfish per al càlcul de jugades. En aquest punt el programa també ha de ser capaç de guiar els motors i moure la peça a la posició indicada, però aquest últim fragment de codi està pendent de programar.

S'ha dissenyat tota la estructura i el sistema mecànic sota la premissa de ser el més econòmic possible i en el cas hipotètic de fabricar-la industrialment sortís molt barata. El sistema mecànic està basat en un sistema cartesià anomenat CoreXY en el qual els motors principals es mantenen fixes permetent així unes millors prestacions dinàmiques. Tot el sistema mecànic està contingut en una caixa de fusta estèticament elegant amb unes dimensions que ofereixen una ergonomia òptima per a un taulell que es situarà sobre una taula.

S'ha dissenyat i fabricat tot el sistema electrònic, que utilitza una Raspberry Pi (un petit ordinador orientat a la robòtica) com a nucli central que allotja i fa funcionar el programa. El sistema electrònic perifèric permet al cervell de la màquina comunicar les ordres del programa als motors a la vegada que rep informació de retroalimentació des de sensors de posició i la càmera.

A la vegada s'ha desenvolupat un sistema de subjecció magnètic que permet moure les peces des de sota del taulell mitjançant imants permanents i un servomotor.

Paraules clau (màxim 10):

CNC	Escacs	Mecatrònica	Stockfish
Anàlisi d'imatge	3D	Motor pas a pas	Raspberry Pi
OpenCV	Python 3		

ABSTRACT

This document shows the approach, procedures and results of a project that aims to design and manufacture a mechatronic system capable of playing chess against a human in a physical way. The system is capable to analyse the situation of the game from an image, from this information calculate its own plays and finally move its own pieces physically.

The computer program is completely written in Python. The main block of the program is designed to extract the board information from a photo taken with a camera, analyse the image and obtain the necessary information to know the game situation. For this process, various functions have been used from the libraries OpenCV, open source library for image analysis and NumPy, open source library for the use of arrays.

With the information extracted from the image analysis it's capable to calculate the plays, for this process various functions of the library Python-Chess had been used to generate and update the virtual field, and also the chess engine Stockfish to calculate the plays. At this point, the machine has to be capable of moving the engines too and move the piece to the indicated location, but this last piece of code is awaiting to be programmed.

All the structure and the mechanical system have been designed under the premise of being as economical as possible and in the hypothetical case of manufacturing it industrially it would be very cheap. The mechanical system is based on a Cartesian system called CoreXY in which the main motors stay static resulting in a better dynamic performance. All the mechanical system is contained in an aesthetically elegant wooden box with an optimum and ergonomic dimensions for a desk that will be placed on a table.

The entire electronic system has been designed and manufactured. It uses a Raspberry Pi (a small robot-oriented computer) as the central core which hosts and operates the program. The peripheral system allows the machine's brain to communicate the program orders to the motors and receive feedback information from position sensors and the camera.

A magnetic fastening system has been developed too, which allows to move the pieces from below the board using permanent magnets and a servomotor.

Keywords (10 maximum):

CNC	Chess	Mechatronics	Stockfish
Image processing	3D	Stepper motor	Raspberry Pi
OpenCV	Python 3		

SUMARI

INTRODUCCIÓ	8
1. ESTAT DE L'ART	10
1.1 Taulells d'escacs electrònics sense moviment físic ni anàlisi d'imatge	10
1.1.1 Arduino & raspberry pi chess computer	10
1.2 Taulells d'escacs amb moviment físic autònom però sense anàlisi d'imatge	11
1.2.1 Square off.....	11
1.3 Taulells d'escacs amb anàlisi d'imatge però sense moviment físic autònom	12
1.3.1 Stonewall	12
1.4 Taulell d'escacs amb anàlisi d'imatge i moviment autònom	13
1.4.1 Raspberry turk	13
1.4.2 Gambit: a robust chess-playing robotic system	14
1.4.3 Robot ajedrecista: sistema mecatrónico aplicado a la toma de decisiones	15
1.4.4 Marineblue: a low-cost chess robot	16
1.4.5 Magnus the chess robot	17
2. TREBALL DE CAMP	18
2.1 Recerca prèvia al pre-disseny	18
2.1.1 Llenguatge de programació	18
2.1.2 Biblioteca i funcions per a l'anàlisi d'imatge.....	18
2.1.3 Llibreria per al treball amb matrius	19
2.1.4 Motor d'escacs.....	19
2.1.5 Taulell virtual	20
2.1.6 El controlador.....	21
2.1.7 GPIO i altres connectors de la Raspberry Pi.....	23
2.1.8 Sistema operatiu	24
2.1.9 Càmera	25
2.1.10 Cable de la càmera	25
2.1.11 Targeta SD	25
2.1.12 Font d'alimentació 5V per a la Raspberry Pi.....	26
2.1.13 Tipus de configuracions mecàniques rellevants	27
2.1.14 Configuració mecànica escollida	29
2.1.15 Sistema de subjecció de les peces	31
2.2 Recerca posterior al pre-disseny	32
2.2.1 Motors pas a pas.....	32
2.2.2 Drivers per als motors.....	32
2.2.3 Font d'alimentació 12V	33
2.2.4 Connector jack-regleta per a font d'alimentació	34
2.2.5 Servomotor	34
2.2.6 Ventiladors.....	34
2.2.7 Potenciòmetres encoders	35
2.2.8 Engranatges per al sistema d'encoders	37
2.2.9 Conversor AD per a la lectura d'encoders	38
2.2.10 Politges i corretges	40
2.2.11 Politges motors.....	41
2.2.12 Politges lliures dentades.....	42
2.2.13 Politges lliures llises	42

2.2.14	Corretja	43
2.2.15	Tensor de corretja	43
2.2.16	Rodes	43
2.2.17	Guies comercials	44
3.	DISSENY I FABRICACIÓ	45
3.1	Disseny del taulell	45
3.2	La elecció de les peces	48
3.3	Disseny del sistema mecànic i estructural	50
3.3.1	Procés de disseny. la caixa.	51
3.3.2	Procés de disseny. El suport del motor.	55
3.3.3	Procés de disseny. Les guies fixes i els seus suports.	57
3.3.4	Procés de disseny. Guia mòbil.	61
3.3.5	Procés de disseny. Suport de les politges en la guia mòbil.	64
3.3.6	Sistema mòbil 2D	67
3.3.7	Procés de disseny. El suport de les politges fixes.	69
3.3.8	Sistema mòbil 3D / Sistema de subjecció magnètic.....	70
3.3.9	Procés de disseny. Estructura per a la càmera i la il·luminació.	72
3.3.10	Procés de disseny. Disseny del suport dels potenciòmetres.	77
3.4	Sistemes elèctric i electrònic	78
3.4.1	Esquema electrònic general. Diagrama de flux.....	78
3.4.2	Esquema elèctric general	79
3.4.3	Esquema de connexions al GPIO de la Raspberry Pi	80
3.4.4	Connexió de la càmera a la Raspberry Pi	80
3.4.5	Esquema de connexions al driver DRV8825 n°0	81
3.4.6	Esquema de connexions al driver DRV8825 n°1	82
3.4.7	Esquema de connexions al potenciòmetre n°0	83
3.4.8	Esquema de connexions al potenciòmetre n°1	83
3.4.9	Esquema de connexions al PIC 12F1822	83
3.4.10	Placa dels drivers	84
3.4.11	Distribuïdor.....	84
3.5	Programa informàtic	85
3.5.1	Diagrama de flux del cos del programa	85
3.5.2	Cos del programa	86
4.	CONCLUSIONS	98
5.	LÍNIES FUTURES DE TREBALL.....	101
	PRESSUPOST.....	102
	CROQUIS FINALS ACOTATS AMB VARIABLES	105
	FULL DE CàLCUL.....	128
	PLÀNOLS	140

FUNCIONS D'ELABORACIÓ PRÒPIA UTILITZADES EN EL PROGRAMA	155
Funció per enquadrar la imatge.....	155
Funció per rotar la imatge.....	156
Funció per detectar les peces	156
Funció per detectar el color	160
Funcions per extrapolar el moviment realitzat	163
Funció de creació del mapa nodal	168
Funció per localitzar la posició de l'actuador	171
Funció buscador de camins.....	174
Funció de moviment dels motors pas a pas.....	174
Funció de moviment del servomotor.....	175
 PROTOTIP DESENVOLUPAT. RECURS D'IMATGES REPRESENTATIVES.	 176
 BIBLIOGRAFIA	 180

Sumari de figures

Figura 1. Foto de taulell. (font: http://chess.forththerapy.co.uk/home/)	10
Figura 2. Taulell d'escacs square off. (font: https://www.kickstarter.com)	11
Figura 3. Captura de la demostració del projecte stonewall. (font: https://www.youtube.com)	12
Figura 4. Foto del raspberry pi turk. (font: http://www.raspberrypitürk.com/)	13
Figura 5. Captura del vídeo de demostració de gambit. (font: https://www.youtube.com)	14
Figura 6. Imatge del projecte "robot ajedrecista: sistema mecatrónico aplicado a la toma de decisiones" (font: http://sedici.unlp.edu.ar/bitstream/handle/10915/58393/documento_completo.pdf-pdf.pdf?sequence=1)	15
Figura 7. Captura del vídeo de demostració del projecte marineblue. (font: https://www.youtube.com)	16
Figura 8. Foto del projecte magnus. (font: https://daim.idi.ntnu.no/masteroppgaver/010/10695/masteroppgave.pdf)	17
Figura 9. Logo python. (font: https://www.python.org/)	18
Figura 10. Logo opencv. (font: https://opencv.org/)	18
Figura 11. Logotip de numpy. (font: http://www.numpy.org/)	19
Figura 12. Logo de stockfish. (font: https://stockfishchess.org/)	20
Figura 13. Capçalera web python-chess. (font: http://python-chess.readthedocs.io/en/latest/)	20
Figura 14. Logotip raspberry pi. (font: https://www.raspberrypi.org/)	22
Figura 15. Raspberry pi 3 model b. (font: https://www.raspberrypi.org/)	23
Figura 16. Esquema dels pins gpio i altres connectors de la raspberry pi	24
Figura 17. Logo de raspbian. (font: https://www.raspbian.org/)	24
Figura 18. Raspberry pi camera module v2. (font: https://www.raspberrypi.org/)	25
Figura 19. Cable d'1m per la càmera. (font: https://www.az-delivery.de/)	25
Figura 20. Targeta microsd de 16 gb. (font: https://www.sandisk.es/)	26
Figura 21. Font 5v 3a micro usb per raspberry pi. (font: https://www.amazon.es)	26
Figura 22. Esquema de la configuració cartesiana. (font: https://www.udesantiagovirtual.cl/wp/)	27
Figura 23. Esquema de la configuració cilíndrica. (font: https://www.udesantiagovirtual.cl/wp/)	27
Figura 24. Esquema de la configuració rotacional. (font: https://www.udesantiagovirtual.cl/wp/)	28
Figura 25. Esquema de la configuració scara. (font: https://www.udesantiagovirtual.cl/wp/)	28
Figura 26. Esquema de la configuració en paral·lel. (font: https://www.udesantiagovirtual.cl/wp/)	29
Figura 27. Logotip de core xy. (font: http://corexy.com/index.html)	30
Figura 28. Esquema core xy. (font: http://corexy.com/theory.html)	30
Figura 29. Foto del motor nema 17. (font: http://www.motionking.com)	32
Figura 30. Esquema motor nema 17. (font: http://www.motionking.com)	32
Figura 31. Foto del driver drv8825. (font: https://www.pololu.com)	33
Figura 32. Esquema de connexions del driver drv8825. (font: https://www.pololu.com)	33
Figura 33. Font d'alimentació 12v 3a. (font: https://www.amazon.es)	33
Figura 34. Connector jack femella a regleta per la font d'alimentació de 12v. (font: https://www.amazon.es)	34
Figura 35. Motor sg90 previst d'utilitzar en el sistema de subjecció magnètica de les peces. (font: http://tienda.bricogeek.com)	34
Figura 36. Ventilador 12v 50x50mm. (font: https://www.amazon.es)	35
Figura 37. Foto del potenciòmetre bourns multivolta. (font: https://www.electronicaembajadores.com)	37
Figura 38. Foto de l'engrenatge a 45° de rs pro. (font: https://es.rs-online.com)	37
Figura 39. Muntatge relatiu entre les dues rodes dentades. (font: https://docs-emea.rs-online.com/webdocs/157a/0900766b8157a694.pdf)	38
Figura 40. Foto del pic 12f1822. (font: http://www.microchip.com)	39
Figura 41. Diagrama de pins del pic12f1822. (font: http://ww1.microchip.com/downloads/en/devicedoc/40001413e.pdf)	39
Figura 43. Vis sense fi per impressora 3d. (font: https://www.amazon.es)	40
Figura 43. Esquema de muntatge del sistema de transmissió amb vis sense fi. (font: elaboració pròpia)	40
Figura 44. Logo de rewrap. (font: http://rewrap.org)	41
Figura 45. Politges gt2 de 20 dents. (font: https://www.amazon.es)	41
Figura 46. Foto de les politges motors. (font: https://www.amazon.es)	41

Figura 47. Politges lliures dentades gt2. (font: https://www.amazon.es)	42
Figura 48. Politges lliures llises gt2. (font: https://www.amazon.es).....	42
Figura 49. Corretja gt2. (font: https://www.3dpenedes.com)	43
Figura 50. Tensor de corretja. (font: https://www.3dpenedes.com).....	43
Figura 51. Esquemes de diferents tipus de rodes. (font: elaboració pròpia)	44
Figura 52. Rodes tipus v-slot. (font: https://www.amazon.es)	44
Figura 53. Tres tipus diferents de guies lineals. (font: https://www.amazon.es)	44
Figura 54. Guia per a roda v-slot. (http://openbuildspartstore.com)	44
Figura 55. Taulell i peces que es preveien utilitzar inicialment. (font: elaboració pròpia).....	45
Figura 56. Detall de les peces i el taulell inicials on 'observen les vetes de la fusta i el poc contrast de color entre els escacs negres i les peces blanques. (font: elaboració pròpia)	46
Figura 57. Esquema inicial del taulell. (font: elaboració pròpia)	46
Figura 58. Imatge del taulell d'escacs. (font: elaboració pròpia)	47
Figura 59. Taulell de fusta contraxapada de 5mm amb el vinil enganxat. Al fons forat en l'estructura principal amb encaix per al taulell. (font: elaboració pròpia).....	47
Figura 60. Vista detallada de les peces d'escacs. (font: elaboració pròpia)	48
Figura 61. Comparativa entre les peces que es preveien utilitzar inicialment i les que s'han acabat utilitzant. (font: elaboració pròpia)	48
Figura 62. Comparativ entre la proporció d'escac ocupat per les peces previstes inicialment i les finals. (font: elaboració pròpia).....	48
Figura 63. Peces amb la tapa inferior posada i treta on es pot veure el muntatge de l'imant permanent interior. (font: elaboració pròpia).....	49
Figura 64. Vista del taulell amb les peces col·locades. (font: elaboració pròpia).....	49
Figura 65. La màquina vista des de sota sense la tapa inferior. (font: elaboració pròpia)	50
Figura 66. Vista general de la estructura. (font: elaboració pròpia).....	50
Figura 67. Pre-disseny de la caixa. Caixa puzzle. (font: elaboració pròpia)	51
Figura 68. Pre-disseny de la caixa. Caixa amb parets de tub. (font: elaboració pròpia)	52
Figura 69. Captura del 3d vista des de sota. (font: elaboració pròpia)	53
Figura 70. El prototip vist des de sota sense la tapa inferior. (font: elaboració pròpia)	53
Figura 71. Captura del 3d, vista superior de la caixa. (font: elaboració pròpia).....	54
Figura 72. Caixa vista des de dalt. (font: elaboració pròpia)	54
Figura 73. Detall d'un vèrtex de la caixa del prototip. (font: elaboració pròpia)	54
Figura 74. Escaires estructurals i d'ajust. (font: elaboració pròpia)	55
Figura 76. Esquema de muntatge dels motors sobre els suports en "I". (font: elaboració pròpia)	55
Figura 76. Suport per al motor nema 17 en "I". (font: https://www.amazon.es)	55
Figura 77. Esquema del disseny del suport del motor exterior i regulable, (font: elaboració pròpia).....	56
Figura 78. Detall del suport del motor collat a l'estructura principal. Vista des de sota. (font: elaboració pròpia)	56
Figura 79. Detall del motor travessant el forat mecanitzat en la estructura principal. (font: elaboració pròpia)	57
Figura 80. Diversos esquemes de models de guia fixe tipus "sandwich". (font: elaboració pròpia).....	57
Figura 81. Esquema de guia en forma d'escaire i sistema d'unió. (font: elaboració pròpia)	58
Figura 82. Esquema de la idea d'utilitzar la pròpia caixa com a guia. (font: elaboració pròpia)	59
Figura 83. Rodes tipus v-slot. (font: https://www.amazon.es)	59
Figura 84. Diverses formes per a les guies fixes. (font: elaboració pròpia).....	59
Figura 85. Esquema de guia en forma de bolet i rodes compatibles. (font: elaboració pròpia)	60
Figura 86. Captura del 3d. Vista d'una guia fixe, els suports i les rodes a sobre. (font: elaboració pròpia)	60
Figura 87. Foto del prototip. Detall de la guia fixe amb les rodes a sobre. (font: elaboració pròpia).....	61
Figura 88. Grapa per tub rodó Ø6 mm. (font: https://www.ferreteriasindustriales.es)	61
Figura 89. Diferents propostes per a la guia mòbil. (font: elaboració pròpia)	62
Figura 90. Diversos plantejaments respecte el muntatge a utilitzar en la guia mòbil. (font: elaboració pròpia)	62
Figura 91. Tub quadrat d'alumini 20x20. (font: http://www.leroymerlin.es)	63
Figura 92. Inserts per tub quadrat de 20x20mm amb forat roscat m8. (font: elaboració pròpia).....	63
Figura 93. Costat dret dels sistema de guia mòbil en el 3d. Vista superior. (font: elaboració pròpia)	63
Figura 94. Foto del costat esquerre de la guia mòbil en el prototip. Vista lateral. (font: elaboració pròpia)	

.....	64
Figura 95. Esquema de muntatge de les politges mòbils. (font: elaboració pròpia)	64
Figura 96. Diversos dissenys per allotjar les politges i unir-les a la guia mòbil i a les rodes de la guia fixe a la vegada que havien de permetre el pas de les corretges. (font: elaboració pròpia)	65
Figura 97. Disseny del suport de les politges mòbils en base a un bloc de fusta mecanitzat en dues parts. (font: elaboració pròpia).....	65
Figura 98. Suport de les politges mòbils amb les politges i les rodes muntades. Captura del 3d. (font: elaboració pròpia).....	66
Figura 99. Foto del suport de les politges mòbils en el prototip. (font: elaboració pròpia).....	66
Figura 100. Vista de l'actuador 2d. Captura del disseny 3d. (font: elaboració pròpia)	68
Figura 101. Vista de l'actuador 2d. Foto del prototip. (font: elaboració pròpia)	68
Figura 102. Placa de suport de les politges fixes amb les politges muntades. (font: elaboració pròpia)...	69
Figura 103. Politges fixes muntades en el prototip (font: elaboració pròpia)	69
Figura 104. Electroimants de 2'5, 5, 10, 20 i 500 n de força consecutivament amb els que es van fer proves de subjecció electromagnètica (font: https://www.amazon.es)	70
Figura 105. Muntatge del servo sobre l'actuador 2d amb l'iman de neodimi a la punta del braç. (font: elaboració pròpia).....	70
Figura 106. Motor sg90 previst d'utilitzar en el sistema de subjecció magnètica de les peces. (font: http://tienda.bricogeek.com)	71
Figura 107. Imants de neodimi de 6x6mm. (font: https://www.superimanes.com)	71
Figura 108. Muntatge provisional inivial per a l'anàlisi d'imatge. (font: elaboració pròpia).....	72
Figura 109. Muntatge provisional versió 2 per a l'anàlisi d'imatge. Detall del muntatge de la càmera. (font: elaboració pròpia).....	73
Figura 110. Muntatge provisional versió 2 per a l'anàlisi d'imatge. Vista del conjunt. (font: elaboració pròpia)	73
Figura 111. Muntatge provisional versió 2 per a l'anàlisi d'imatge. Detall de la il·luminació del taulell. (font: elaboració pròpia).....	74
Figura 112. Muntatge de la càmera i el sistema d'il·luminació final en el prototip. Vista des de sota. (font: elaboració pròpia).....	74
Figura 113. Vista general del disseny 3d de la estructura. (font: elaboració pròpia)	75
Figura 114. Vista general del prototip. (font: elaboració pròpia).....	75
Figura 115. Detall del muntatge de la càmera en el prototip. (font: elaboració pròpia)	76
Figura 116. Detall de la unió entre la estructura de la càmera i la estructura principal. (font: elaboració pròpia)	76
Figura 117. Detall de la sortida dels cables provinents de les làmpades i la càmera a través de la estructura principal. Vista des de sota. (font: elaboració pròpia)	76
Figura 118. Vista del suport del potenciòmetre en el 3d. (font: elaboració pròpia)	77
Figura 119. Vista del suport del potenciòmetre en el prototip. (font: elaboració pròpia)	77
Figura 120. Esquema electrònic general. Diagrama de flux. (font: elaboració pròpia)	78
Figura 121. Esquema elèctric. (font: elaboració pròpia)	79
Figura 122. Esquema de connexions al gpio de la raspberry pi. (font: elaboració pròpia)	80
Figura 123. Connexió de la càmera a la raspberry pi. (font: https://www.zapals.com).....	80
Figura 124. Esquema de connexions al driver nº 0. (font: elaboració pròpia)	81
Figura 125. Esquema de connexions al driver nº 1. (font: elaboració pròpia)	82
Figura 126. Esquema de connexions al potenciòmetre nº0. (font: elaboració pròpia).....	83
Figura 127. Esquema de connexions al potenciòmetre nº1. (font: elaboració pròpia).....	83
Figura 128. Esquema de connexions al pic12f1822. (font: elaboració pròpia)	83
Figura 129. Placa on es connecten els drivers. (font: elaboració pròpia).....	84
Figura 130. Esquema del distribuïdor. (font: elaboració pròpia)	84
Figura 131. Diagrama de flux simplificat del cos del programa. (font: elaboració pròpia)	85
Figura 132. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc. (font elaboració pròpia).....	87
Figura 133. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc. (font elaboració pròpia).....	89
Figura 134. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc. (font elaboració pròpia).....	90
Figura 135. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc.	

(font elaboració pròpia).....	91
Figura 136. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc. (font elaboració pròpia).....	92
Figura 137. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc. (font elaboració pròpia).....	93
Figura 138. Esquemes manuals geomètrics principals. (font: elaboració pròpia).....	106
Figura 139. Esquema manual del contacte suport_guia_fixe - guia_fixe - roda. (font: elaboració pròpia)	107
Figura 140. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	108
Figura 141. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	109
Figura 142. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	110
Figura 143. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	111
Figura 144. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	112
Figura 145. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	113
Figura 146. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	114
Figura 147. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	115
Figura 148. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	116
Figura 149.. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	117
Figura 150. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	118
Figura 151. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	119
Figura 152. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	120
Figura 153. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	121
Figura 154. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	122
Figura 155. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	123
Figura 156. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	124
Figura 157. Esquemes manuals acotats amb variables. (font: elaboració pròpia).....	125
Figura 158. Esquemes manuals acotats amb valors. (font: elaboració pròpia).....	126
Figura 159. Esquemes manuals acotats amb valors. (font: elaboració pròpia).....	127
Figura 160. Transformació de perspectiva. (font: cv2.warperspective).....	155

GLOSSARI DE SIGNES, SÍMBOLS, ABREVIATURES, ACRÒNIMS I TERMES

Driver	Dispositiu electrònic a través del qual un microcontrolador o microprocessador pot controlar un motor. Fa de pont entre la electrònica de senyal i la electrònica de potència.
Encoder	És un sensor rotatiu que permet captar l'angle girat.
Microcontrolador	Circuit integrat que conté una CPU, RAM, ROM, ports d'entrada i sortida i perifèrics connectats entre sí en l'interior de l'encapsulat.
Microprocessador	Contenen únicament la CPU i per funcionar requereixen una instal·lació electrònica addicional de memòria i altres perifèrics. Acostumen a tenir molta més potència de processament que els microcontroladors.
Anàlisi d'imatge	Es basa en l'aplicació d'algoritmes sobre la matriu de valors que compon la imatge per extreure'n patrons.
Intel·ligència artificial	Algoritmes matemàtics aplicats a la informàtica que són capaços d'aïllar patrons de forma autònoma a partir del processament de grans quantitats de dades.
Visió artificial	Intel·ligència artificial aplicada a l'anàlisi d'imatge.
Motor d'escacs	Programa informàtic capaç de calcular una jugada d'escacs.
Taulell d'escacs virtual	Tècnicament GUI (Graphical User Interface) permet la comunicació UCI amb el motor d'escacs i la comunicació de la persona amb el joc. A la vegada emmagatzema els estats del taulell i valida les jugades.
Selective Compliance Articulated Robot Arm (SCARA):	Braç robot que es pot moure lliurement en les direccions X i Y però està fixat en la Z
Actuador	Sistema que genera una acció mecànica a partir de la energia que se l'hi subministri, en el cas dels motors elèctrics energia elèctrica.
Motor pas a pas	És un dispositiu electromecànic que transforma impulsos elèctrics en moviments angulars discrets.
Servomotor	És un dispositiu electromecànic compost per un motor de corrent continu unit a una sèrie d'engranatges reductors. La posició de l'eix de sortida és mesurada per un potenciòmetre que envia la informació a un microcontrolador aquest controla a la vegada el gir del motor segons l'impuls que se l'hi dona a través del port de control. Aquest motor només pot girar 180°.
Mecatrònica	Combinació de l'enginyeria mecànica, elèctrica, electrònica i de programació que permet crear sistemes més versàtils i econòmics que utilitzant cada una de les branques per separat.
Robot	Màquina automàtica capaç de fer una sèrie de tasques.
Joc d'escacs	Joc de taula en que dos jugadors competeixen sobre un

	taulell de 8x8 caselles. Cada jugador comença amb 16 peces i guanya qui aconsegueix el mat del rei contrari.
Escac	Casella del taulell d'escacs. El taulell d'escacs esta compost per una matriu de 8x8 escacs.

INTRODUCCIÓ

OBJECTIUS GENERALS DEL PROJECTE

Aquest projecte respon a l'objectiu personal de posar en pràctica alguns dels coneixements assolits a la carrera i desenvolupar-ne d'altres que em despertaven interès i volia conèixer millor.

L'objectiu del projecte no era la maquina en sí, que també, però el que buscava fonamentalment era el procés d'aprenentatge que comportava i els coneixements que hauria assolit al final d'aquest procés.

Els coneixements que volia assolir i que representen els objectius generals del projecte son:

- Millorar els meus coneixements habilitat en la programació informàtica. La programació informàtica la vaig descobrir en una assignatura dins de la carrera d'enginyeria mecànica i em va enganxar. Tenia ganes d'aplicar-la en un projecte gran i em va semblar que el TFG era una bona oportunitat.
- Aprendre més sobre els sistemes electrònics necessaris per a crear el pont entre els programes informàtics i el món físic, podent així unir els dos camps que més m'interessen, l'enginyeria mecànica i la programació informàtica de control.
- Millorar els meus coneixements sobre els mecanismes i el funcionament de les màquines CNC, molt utilitzades en la robòtica industrial i l'automatització de processos.
- Obtenir la experiència de dissenyar una màquina completa des de zero, des del disseny estructural fins al programa informàtic entrant en els camps de la mecànica, el disseny industrial, la electrònica i la informàtica.
- Aprendre les bases de l'anàlisi d'imatge per més endavant entrar a descobrir la visió artificial, un subcamp de la intel·ligència artificial que em sembla molt interessant. L'anàlisi d'imatge i la visió artificial ofereixen moltes possibilitats a l'hora de captar informació i cada vegada s'està utilitzant i desenvolupant més, sobretot en la automatització de processos industrials i en la robòtica autònoma.
- Aprendre a espavilar-me per buscar i estructurar tota la informació necessària per tirar endavant un projecte que no havia fet mai abans.
- Gestionar un projecte que engloba diverses àrees de l'enginyeria que conviuen amb l'enginyeria mecànica en el món industrial, aconseguint així millorar el meu coneixement sobre elles i la manera en que es relacionen.

En resum volia preparar-me per el món real de la enginyeria i donar-me un petit bany de realitat abans d'entrar al món laborat, posar-me en un procés que em superés per sortir-ne amb la seguretat de saber que em podria espavilar en feines exigents i de responsabilitat on hagués de prendre decisions.

A la vegada també volia aprofitar per entrar en temes que m'interessaven i que encara no havia tingut la possibilitat de descobrir amb profunditat.

PERQUÈ UN TAULELL D'ESCACS AUTOMÀTIC AMB ANÀLISI D'IMATGE?

Fa entre un i dos anys em vaig trobar per casualitat un vídeo on es feia una demostració de les capacitats d'un robot autònom creat per Boston Dynamics. Aquell robot em va impressionar i vaig pensar que en un futur m'agradaria poder participar en un projecte similar i crear un robot d'aquell tipus.

Quan l'hi vaig comentar al tutor que volia fer un robot autònom amb visió artificial, d'alguna manera em va dir que em cal més i que simplifiqués el projecte, perquè un projecte que sembla complicat quan et poses a fer-lo et trobes en que es torna extremadament complicat.

Així que entre els dos vam reduir el concepte de robot autònom amb visió artificial a una màquina que fos capaç de jugar a un joc de taula en un taulell físic movent les peces físicament i captant la situació del joc amb una càmera.

Recordo que llavors l'hi vaig preguntar: "però això no és massa senzill per un TFG?" i ell em va mirar amb cara de "no tens ni idea del que t'espera" i assenyalant una impressora que hi havia al despatx em va dir més o menys això: "tu creus que per dissenyar aquesta impressora no hi ha hagut un equip d'enginyers treballant un munt d'hores per dissenyar cada un dels elements que la componen i el sistema en que s'uneixen entre ells per que funcioni com ha de funcionar?", "dissenyar una màquina és un procés molt laboriós i per dissenyar una màquina que jugui a escacs i mogui les peces hi ha una feina".

Jo me'l vaig mirar amb cara de no creure-m'ho gaire i ell va respondre: "de veritat, és molta feina". En aquell moment vaig dir: "d'acord doncs faré això", mentre pensava: "aquest projecte és molt interessant però te pinta que amb dos mesos estarà acabat i quan el presenti el tribunal em dirà que això no és feina suficient per un TFG", però veia el Joan molt convençut així que a mitges m'ho vaig creure i vaig pensar que a males si veia que acabava aviat i em sobrava temps sempre podia ampliar el mateix projecte i així aconseguir més nota.

A dia d'avui he de reconèixer que no tenia ni idea del que significa dissenyar una màquina sencera ni la quantitat de feina i temps que comporta. Després de 8 mesos treballant-hi encara em falten dues setmanes (quatre veient com van les coses) perquè la màquina funcioni com estava previst.

Puc assegurar que la percepció del temps, les etapes i els terminis són unes de les coses més importants que he après en aquest projecte, dissenyar una màquina o un programa informàtic quan es vol portar a la pràctica és un procés extremadament laboriós en el que creus que acabaràs una etapa en dos dies i quan l'acabes ha passat una setmana i mitja.

Els petits detalls que poden semblar banals o que ni et plantejes abans de posar-te a dissenyar són els que més alenteixen el procés, de manera que quan vols preveure el temps que trigaràs a fer un disseny, si no tens experiència en el tipus de projecte en qüestió, el millor és multiplicar la previsió per 3 o per 4 i així potser t'acostaràs al temps real necessari.

Malgrat això s'han assolit la gran majoria dels objectius del projecte i únicament queden pendents de programar tres funcions per al moviment dels motors i adaptar-ne dues més a les últimes característiques del disseny. La resta, tant el programa informàtic principal com la estructura, el sistema mecànic i el sistema electrònic han estat completats i estan preparats per funcionar.

1. ESTAT DE L'ART

Hi ha diversos projectes que es poden trobar escampats per internet en formats molt diversos i amb una quantitat d'informació i estructuració molt variada, en general molt vaga. En aquest apartat es recopilen diversos projectes similars al que es realitza en aquest treball.

La gran majoria són projectes acadèmics. Només n'he trobat un que s'hagi creat amb l'objectiu de ser comercialitzat.

1.1 TAULELLS D'ESCACS ELECTRÒNICS SENSE MOVIMENT FÍSIC NI ANÀLISI D'IMATGE

Els taulells d'escacs electrònics són taulells d'escacs físics que són capaços de detectar les peces en la seva superfície, calcular les seves pròpies jugades i indicar al jugador humà on vol moure la seva peça.

Aquest tipus de taulells existeixen des de fa molts anys i són abundants en el mercat.

Un exemple interessant que utilitza Raspberry Pi i Stockfish és el següent:

1.1.1 ARDUINO & RASPBERRY PI CHESS COMPUTER



Figura 1. Foto de taulell. (Font: <http://chess.forththerapy.co.uk/home/>)

CARACTERÍSTICA	DESCRIPCIÓ
Nom del projecte	Arduino & Raspberry Pi Chess Computer
Web de referència	http://chess.forththerapy.co.uk/home/
Autor	Max (Sense més informació)
Sistema de detecció de peces	Sensors d'efecte hall
Comunicació amb el jugador	LEDs sobre el taulell
Sistema de moviment	No
Motor d'escacs	Stockfish
Hardware de processament	Raspberry Pi
Lloc de creació	USA
Comercialització	No

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

1.2 TAULELLS D'ESCACS AMB MOVIMENT FÍSIC AUTÒNOM PERÒ SENSE ANÀLISI D'IMATGE

1.2.1 SQUARE OFF

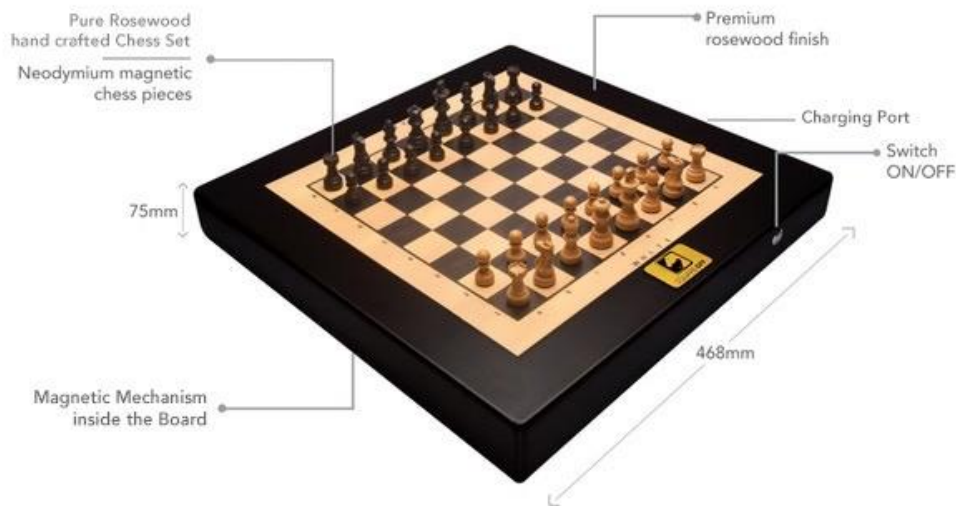


Figura 2. Taulell d'escacs Square Off. (Font: <https://www.kickstarter.com>)

CARACTERÍSTICA	DESCRIPCIÓ
Nom del projecte	Square Off
Web de referència	https://squareoffnow.com/
Autors	<ul style="list-style-type: none">• Aathur Mehta (Co-Fundador)• Bhavya Gohil (Co-Fundador)• Tarun Gupta (Consultor de Crowdfunding)• Alessandro Dionna (Enginyer)• Kevin Zatakia (Executiu de màrqueting)• Deep Latzia (Desenvolupador)
Sistema de detecció de peces	(Sense informació)
Comunicació amb el jugador	Moviment físic A través de Smartphone
Sistema de moviment	Des de sota el taulell. (Sense més informació)
Subjecció de peces	(Magnètic)
Motor d'escacs	(Sense informació)
Hardware de processament	(Sense informació)
Lloc de creació	Índia
Comercialització	Sí

1.3 TAULELLS D'ESCACS AMB ANÀLISI D'IMATGE PERÒ SENSE MOVIMENT FÍSIC AUTÒNOM

1.3.1 STONEWALL

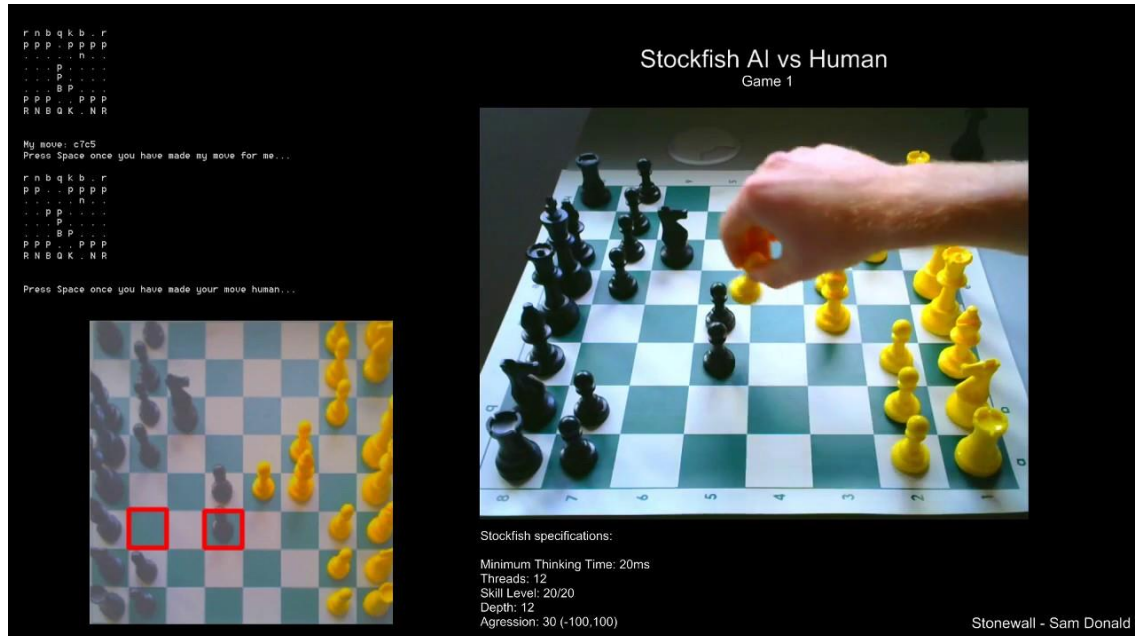


Figura 3. Captura de la demostració del projecte Stonewall. (Font: <https://www.youtube.com>)

CARACTERÍSTICA	DESCRIPCIÓ
Nom del projecte	Stonewall
Web de referència	https://www.youtube.com/watch?v=uDw_vCNbx-I
Autor	Sam Donald
Sistema de detecció de peces	Anàlisi d'imatge (Open CV)
Comunicació amb el jugador	Per pantalla
Sistema de moviment	No
Sistema de subjecció de peces	No
Motor d'escacs	Stockfish
Hardware de processament	(Sense informació)
Llenguatge de programació	Python
Lloc de creació	University of Canterbury (USA)
Comercialització	No

1.4 TAULELL D'ESCACS AMB ANÀLISI D'IMATGE I MOVIMENT AUTÒNOM

1.4.1 RASPBERRY TURK

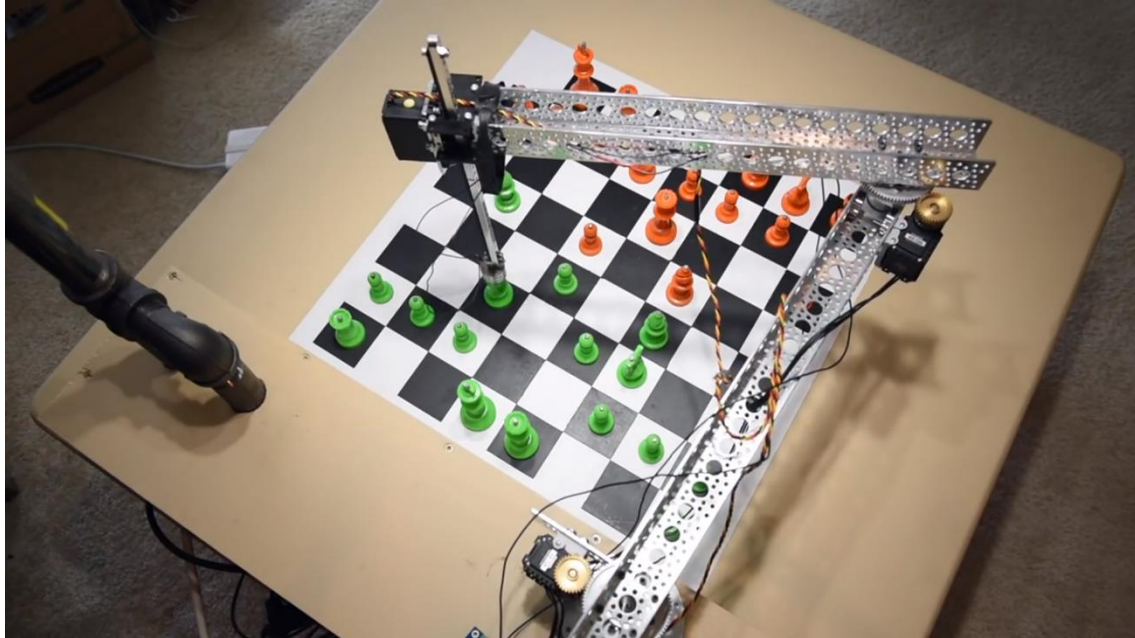


Figura 4. Foto del Raspberry Pi Turk. (Font: <http://www.raspberryturk.com/>)

CARACTERÍSTICA	DESCRIPCIÓ
Nom del projecte	Raspberry Turk
Web de referència	http://www.raspberryturk.com/
Autor	Joey Meyer
Sistema de detecció de peces	Anàlisi d'imatge (OpenCV) i intel·ligència artificial
Comunicació amb el jugador	Moviment físic
Sistema de moviment	Selective Compliance Articulated Robot Arm (SCARA) Motors: servomotors
Sistema de subjecció de peces	Mecanisme mòbil en direcció Z Electroimant al final del braç Imants permanents sobre les peces
Motor d'escacs	Stockfish
Hardware de processament	Raspberry Pi
Llenguatge de programació	Python
Lloc de creació	MIT (USA)
Comercialització	No

1.4.2 GAMBIT: A ROBUST CHESS-PLAYING ROBOTIC SYSTEM

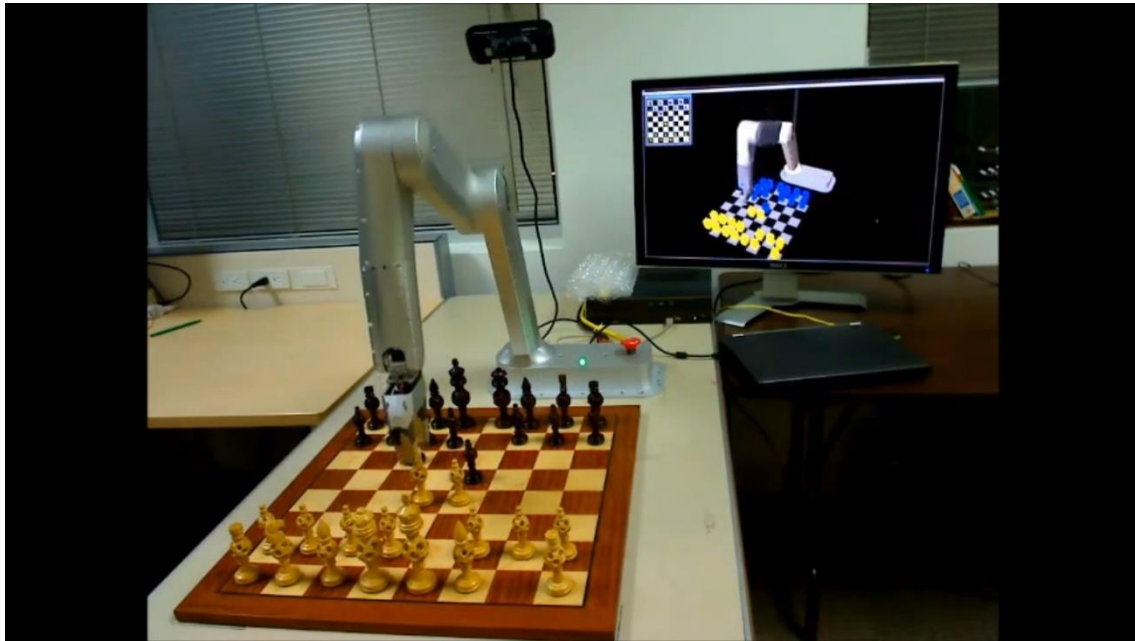


Figura 5. Captura del vídeo de demostració de Gambit. (Font: <https://www.youtube.com>)

CARACTERÍSTICA	DESCRIPCIÓ
Nom del projecte	Gambit: A Robust Chess-Playing Robotic System
Web de referència	https://rse-lab.cs.washington.edu/postscripts/chess-icra-11.pdf
Autors	<ul style="list-style-type: none">• Cynthia Matuszek• Brian Mayton• Roberto Aimi• Marc Peter Deisenroth• Liefeng Bo,• Robert Chu• Mike Kung• Louis LeGrand• Joshua R. Smith• Dieter Fox
Sistema de detecció de peces	Anàlisi d'imatge i visió artificial
Comunicació amb el jugador	Moviment físic i pantalla
Sistema de moviment	Braç articulat
Sistema de subjecció de peces	Pinça paral-lela
Motor d'escacs	(Sense informació)
Hardware de processament	Intel Atom net-top PC
Llenguatge de programació	(Sense informació)
Lloc de creació	UW Robotics and State Estimation Lab (USA)
Comercialització	Si (no per al públic general)
Sensors de l'actuador	Encoders a les articulacions juntament amb PrimeSense
Càmera	Càmera tecnològicament idèntica a una Xbox Kinect

1.4.3 ROBOT AJEDRECISTA: SISTEMA MECATRÓNICO APLICADO A LA TOMA DE DECISIONES



Figura 6. Imatge del projecte "Robot ajedrecista: sistema mecatrónico aplicado a la toma de decisiones" (Font: http://sedici.unlp.edu.ar/bitstream/handle/10915/58393/Documento_completo.pdf-PDFA.pdf?sequence=1)

CARACTERÍSTICA	DESCRIPCIÓ
Nom del projecte	Robot ajedrecista: sistema mecatrónico aplicado a la toma de decisiones
Web de referència	http://sedici.unlp.edu.ar/bitstream/handle/10915/58393/Documento_completo.pdf-PDFA.pdf?sequence=1
Autors	<ul style="list-style-type: none">• Guillermo Larregay• Federico Pinna1• Luis Avila• Daniel Morán
Sistema de detecció de peces	Anàlisi d'imatge (OpenCV) i visió artificial
Comunicació amb el jugador	Moviment físic
Sistema de moviment	Braç robot manipulador de tipus industrial (comercial)
Sistema de subjecció de peces	Pinça
Motor d'escacs	GNU Chess
Hardware de processament	Unidad central SUCC (PC Windows) Controlador específic del robot
Llenguatge de programació	C
Lloc de creació	Laboratorio de Mecatrónica de la Facultad de Ingeniería de la Universidad Nacional de San Luis (Argentina)
Sensors de l'actuador	(Sense informació)

1.4.4 MARINEBLUE: A LOW-COST CHESS ROBOT

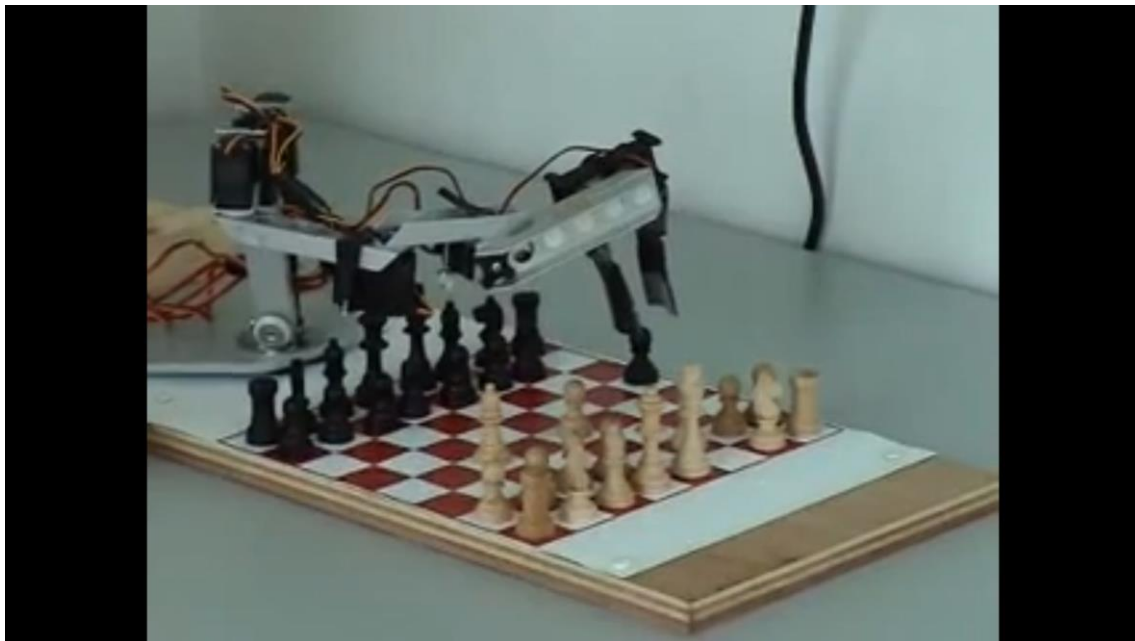


Figura 7. Captura del vídeo de demostració del projecte MarineBlue. (Font: <https://www.youtube.com>)

CARACTERÍSTICA	DESCRIPCIÓ
Nom del projecte	MarineBlue: A Low-Cost Chess Robot
Web de referència	https://pdfs.semanticscholar.org/656d/64d345d252208a40cb1068280d16ed227619.pdf
Autors	<ul style="list-style-type: none">• David Urting• Yolande Berbers
Sistema de detecció de peces	Visió artificial
Comunicació amb el jugador	Moviment físic
Sistema de moviment	Braç robot Robix RC-6
Sistema de subjecció de peces	Pinça
Motor d'escacs	GNU GPL Chessterfield
Hardware de processament	PC (Windows 2000)
Llenguatge de programació	C/C++
Lloc de creació	KU Leuven (Bèlgica)
Sensors de l'actuador	Els encoders dels servos
Càmera	Sony DFW-VL500

1.4.5 MAGNUS THE CHESS ROBOT

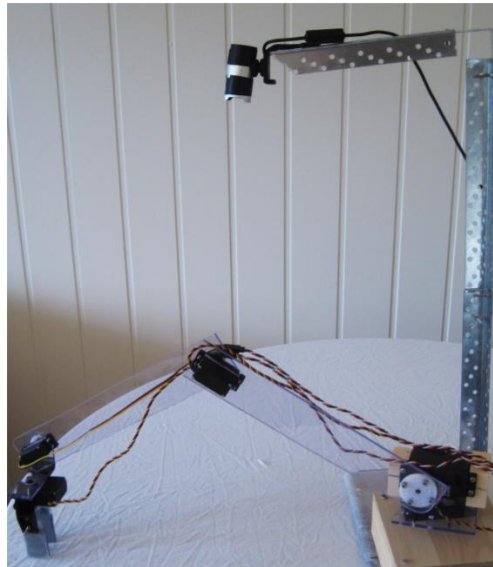


Figura 8. Foto del projecte Magnus. (Font: <https://daim.idi.ntnu.no/masteroppgaver/010/10695/masteroppgave.pdf>)

CARACTERÍSTICA	DESCRIPCIÓ
Nom del projecte	Magnus the Chess Robot
Web de referència	https://daim.idi.ntnu.no/masteroppgaver/010/10695/masteroppgave.pdf
Autors	Tobias Helstad Unger
Sistema de detecció de peces	Anàlisi d'imatge (OpenCV)
Comunicació amb el jugador	Moviment físic
Sistema de moviment	Braç articulats
Sistema de subjecció de peces	Pinça
Motor d'escacs	Chenard
Hardware de processament	(Sense informació)
Llenguatge de programació	C++
Lloc de creació	Norwegian University of Science and Technology (Noruega)
Sensors de l'actuador	Encoders dels servos
Càmera	Microsoft LifeCam Studio

2. TREBALL DE CAMP

2.1 RECERCA PRÈVIA AL PRE-DISENY

2.1.1 LLENGUATGE DE PROGRAMACIÓ

Inicialment tenia la idea de programar en Android per fer funcionar el programa en un Smartphone i així estalviar-me el processador, la càmera, la pantalla i els botons. El problema era que per poder programar en Android primer n'havia d'aprendre. El fet és que no tenia inconvenients en aprendre u nou llenguatge de programació, però hagués gastat massa temps en el procés i el temps era molt limitat en comparació a l'envergadura del projecte.

Feta aquesta reflexió vaig considerar que si era possible fer el programa en Python, que era el llenguatge de programació que coneixia i amb el que ja tenia pràctica, seria la millor opció.



Figura 9. Logo Python. (Font: <https://www.python.org/>)

Vaig buscar informació sobre projectes similars al que jo volia fer, que incloguessin anàlisi d'imatge o motors d'escacs compatibles amb Python i en vaig trobar alguns. Vaig buscar informació sobre els motors d'escacs i les biblioteques que utilitzaven i efectivament existien, eren compatibles amb Python i hi havia molta informació disponible, ja que seguien la filosofia Open Source.

Python és un llenguatge de programació interpretat, interactiu i orientat a objectes. Incorpora mòduls, excepcions, tipografia dinàmica, tipus de dades dinàmiques de molt alt nivell i classes. Python combina un poder notable amb una sintaxi molt clara. Té interfícies per a moltes crides i biblioteques del sistema, així com per a diversos sistemes de finestres, i és extensible en C o C++. També es pot utilitzar com a llenguatge d'extensió per a aplicacions que necessiten una interfície programable. Finalment, Python és portàtil: s'executa en moltes variants Unix, en Mac i Windows 2000 i posteriors.

Web oficial de Python: <https://www.python.org/>

2.1.2 BIBLIOTECA I FUNCIONS PER A L'ANÀLISI D'IMATGE

Per a l'anàlisi d'imatge en entorn Python vaig decidir utilitzar la biblioteca OpenCV (Open Source Computer Vision Library).



Figura 10. Logo OpenCV. (Font: <https://opencv.org/>)

OpenCV (Open Source Computer Vision Library) es publica sota una llicència BSD i, per

tant, és gratuïta per a ús acadèmic i comercial. Compta amb interfícies de C ++, Python i Java i suporta Windows, Linux, Mac OS, iOS i Android. OpenCV va ser dissenyat per a l'eficiència computacional i amb un fort enfocament en aplicacions en temps real. Escrit en C/C++ optimitzat, la biblioteca pot aprofitar el processament de diversos nuclis.

Adoptat a tot el món, OpenCV té més de 47 mil persones de comunitat d'usuaris i la quantitat estimada de descàrregues supera els 14 milions. Els intervals d'ús van des de l'art interactiu, fins a la inspecció de mines, costant mapes a la xarxa o mitjançant robòtica avançada.

Es va decidir utilitzar OpenCV per diversos motius:

- Segueix la filosofia Open Source (Es pot utilitzar lliurement, gratuïtament i es pot trobar molta informació per aprendre)
- Existeixen una gran quantitat de manuals i tutorials creats per la comunitat enfocats directament a OpenCV-Python.
- Gran quantitat de funcions adaptades al llenguatge Python
- Compatible amb Linux (condició necessària per utilitzar-lo en el sistema operatiu Raspbian de la Raspberry Pi)

Web oficial d'OpenCV: <https://opencv.org/>

2.1.3 LLIBRERIA PER AL TREBALL AMB MATRIUS

NumPy és una biblioteca enfocada a la informàtica científica programada en Python que permet treballar amb matrius homogènies multidimensionals.



Figura 11. Logotip de NumPy. (Font: <http://www.numpy.org/>)

A més dels seus usos científics evidents, NumPy també es pot utilitzar com un contenidor multidimensional eficient de dades genèriques.

A part d'aquestes qüestions, NumPy es un requisit indispensable per poder treballar amb la biblioteca OpenCV en Python, ja que les imatges són matrius de valors de grans dimensions i OpenCV requereix disposar de la biblioteca NumPy per gestionar-les de forma eficient.

Web oficial de NumPy: <http://www.numpy.org/>

2.1.4 MOTOR D'ESCACS

El motor d'escacs es el programa informàtic capaç de calcular una jugada d'escacs a partir d'una situació donada del taulell.

Programar un motor d'escacs és una tasca enorme que requereix uns elevats coneixements de programació, lògica, càlcul i òbviament coneixements d'escacs entre d'altres qüestions. Programar un motor d'escacs donaria per un TFG sencer i no crec que em convalidés un TFG d'enginyeria mecànica.

Si algú esta interessat en el funcionament intern dels motors d'escacs pot consultar la següent web, però no és necessari per a realitzar aquest projecte:

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

<https://chessprogramming.wikispaces.com/>

Degut a la gran dificultat de crear un motor d'escacs propi vaig utilitzar un motor d'escacs creat per experts anomenat Stockfish. Un motor força famós, com a mínim en el "mundillo" dels escacs i la intel·ligència artificial ja que és un dels motors d'escacs més potents del mon.



Figura 12. Logo de Stockfish. (Font: <https://stockfishchess.org/>)

Stockfish és de codi obert a diferència de la majoria dels motors d'escacs (llicència GPL). Això vol dir que es pot llegir el codi, modificar-lo, contribuir-hi i utilitzar-lo en projectes.

Es compatible amb una gran quantitat de plataformes: Windows, OS X, Linux, iOS i Android. Això significa que és compatible amb aquest projecte, ja que es realitzarà en una Raspberry Pi, que utilitza Raspbian com a sistema operatiu. Raspbian esta basat en Debian, un sistema operatiu de Linux.

Web oficial de Stockfish: <https://stockfishchess.org/>

Web on es pot consultar el funcionament tècnic intern de Stockfish:

<https://chessprogramming.wikispaces.com/Stockfish>

2.1.5 TAULELL VIRTUAL

Per a generar el taulell virtual d'escacs, generar moviments i comprovar si les jugades son vàlides s'ha utilitzat la biblioteca Python-Chess.

Python-chess és una llibreria purament d'escacs en llenguatge Python. Aquesta llibreria permet entre d'altres:

- Generar un taulell virtual on es representen les diverses peces.
- Es poden moure les peces a través d'un codi enviat a través del terminal o de forma interna en el programa que creï cadascú.
- Conté un sistema de validació de jugades que comprova si les jugades son legals, pseudo-legals o il·legals.

 **python-chess: a pure Python chess library**

build passing coverage 90% pypi package 0.23.5 docs passing

Figura 13. Capçalera web Python-Chess. (Font: <http://python-chess.readthedocs.io/en/latest/>)

A part d'això disposa de funcions específiques per a la comunicació UCI (Universal Chess Interface) amb el motor d'escacs, i és perfectament compatible amb el motor d'escacs Stockfish utilitzat en aquest projecte i comentat anteriorment.

2.1.6 EL CONTROLADOR

A l'hora de dissenyar quin cervell utilitzar per a la màquina em vaig trobar davant de diverses possibilitats:

Utilitzar un ordinador PC, de sobretaula o portàtil:

D'aquesta manera disposaria d'una gran capacitat de processament a la vegada que hi podria connectar perifèrics com ara pantalla, teclat, ratolí, web cam, etc. Amb aquest sistema podria programar directament en l'ordinador que faria de cervell de la màquina i així poder fer modificacions de manera molt més ràpida, a la vegada que podria visualitzar el procés per pantalla i així detectar errors en el programa més fàcilment.

A part podria programar en qualsevol llenguatge, entre ells Python que era el llenguatge que volia utilitzar.

L'inconvenient era que volia que fos un dispositiu compacte i de baix cost, i afegir un ordinador elevava moltíssim aquest cost. Es podria suposar que jo ja disposava d'un ordinador i per tant es podria excloure del pressupost, però la idea era que fos una màquina completa i que portés incorporats tots els elements necessaris per al seu funcionament. A part d'això el cervell també s'havia de connectar amb els actuadors (motors) i sensors (potenciòmetres) cosa que un PC no pot fer directament, necessitaria electrònica auxiliar, com ara un Arduino, que també elevava el cost.

Una altra opció era utilitzar un telèfon mòbil:

Utilitzant un telèfon mòbil podria suposar que molta gent el té i per tant es podria eliminar del pressupost i a la vegada és un dispositiu compacte que es podria incorporar temporalment a la màquina. També tenia altres beneficis com ara:

- La pantalla tàctil que permetria interactuar amb la màquina sense necessitat de perifèrics addicionals
- Una càmera, que es podria utilitzar per captar les imatges.
- Un processador molt potent capaç de fer funcionar el programa.

El gran problema que se'm plantejava era la necessitat d'aprendre a programar en Android i no disposava del temps per fer-ho, així que vaig seguir buscant.

La tercera possibilitat era utilitzar un Arduino:

Arduino es una placa basada en el microcontrolador ATmega328P. Els beneficis que comportaria fer servir un microcontrolador són la facilitat per a comunicar-se amb sensors i actuadors. Els grans problemes serien:

- Un microcontrolador està preparat per fer una tasca relativament simple per a la qual es programa, No té la capacitat per controlar a la vegada tot el que aquesta màquina havia de controlar.
- No se l'hi pot connectar una pantalla per veure el procés que segueix el programa.
- No es pot programar directament sobre ell, s'ha de programar en un PC i després carregar-li el programa, de manera que el procediment de prova i error es s'alenteix.

- S'ha de programar en llenguatge C, llenguatge que no conec i que hauria d'haver après amb un temps del que no disposava.

La “última possibilitat” era fer servir una Raspberry Pi:

Escriu “última possibilitat” entre cometes perquè segur que algú que sàpiga més d'informàtica trobaria altres maneres de fer-ho, com per exemple muntar-se la seva pròpia placa o combinar diferents dispositius, però amb els coneixements limitats d'informàtica que jo tinc aquestes eren les possibilitats que veia més clares i factibles.

Raspberry Pi és un ordinador de placa única de mida d'una targeta de crèdit. Malgrat ser de mida reduïda té una gran capacitat i es pot utilitzar tant en projectes d'electrònica com en les típiques tasques d'un PC d'escriptori. Per exemple fulls de càlcul, processament de textos, navegació per Internet, jocs o reproduir vídeo en alta definició.

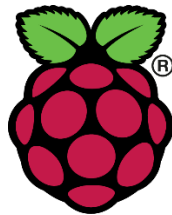


Figura 14. Logotip Raspberry Pi. (Font: <https://www.raspberrypi.org>)

La Raspberry m'oferia tot el necessari per a aquest projecte:

- Em permetia incorporar el cervell a l'interior de la màquina seguint la idea de crear una màquina compacta.
- Com que és un petit ordinador i funciona com a tal, l'hi podia connectar una pantalla i altres perifèrics per programar directament sobre ell.
- Es pot programar en qualsevol llenguatge.
- Disposa d'un mòdul específic de càmera.
- Disposa d'una gran quantitat de pins GPIO on connectar sensors i actuadors.
- És barata.
- Al funcionar amb un sistema operatiu basat en Linux hi havia un munt de llibreries compatibles que podia utilitzar en el programa.
- Té una bona capacitat de processament. Amb una CPU Quad Core 1.2GHz Broadcom BCM2837 de 64 bits i 1 GB de RAM
- Segueix la filosofia Open Source i disposa d'una gran quantitat d'informació disponible a la xarxa i d'una comunitat molt activa.

L'únic inconvenient que presentava era la impossibilitat de llegir entrades analògiques directament, però es va resoldre instal·lant un microcontrolador per traduir la senyal.

La Raspberry Pi específica utilitzada en aquest projecte és el model B de la 3a generació. Aquesta placa és el centre on es processa tota la informació, és la que gestiona i interpreta el programa informàtic escrit en la targeta SD i envia les ordres als diferents perifèrics.



Figura 15. Raspberry Pi 3 Model B. (Font: <https://www.raspberrypi.org>)

Web principal de Raspberry Pi: <https://www.raspberrypi.org/>

Documentació: <https://www.raspberrypi.org/documentation/>

Datasheet:

https://www.raspberrypi.org/documentation/hardware/computemodule/datasheets/rpi_DATA_CM_1p0.pdf

2.1.7 GPIO I ALTRES CONNECTORS DE LA RASPBERRY PI

Els connectors de que disposa la Raspberry són els següents:

- BCM43438 LAN sense fils molt útil per connectar-se a internet i utilitzar la Raspberry per navegar per internet i buscar informació o traspassar informació entre ordinadors.
- Bluetooth Low Energy (BLE) (no utilitzat)
- GPIO de 40 pins on s'han connectat els drivers que controlen els motors, el microcontrolador que tradueix les senyals del potenciòmetres i el servomotor.
- 4 ports USB 2, dos dels quals s'han utilitzat per connectar un teclat i un ratolí.
- Sortida estèreo de 4 pols i port de vídeo compost (no utilitzat)
- HDMI on s'ha connectat una pantalla.
- Port de càmera CSI on s'ha connectat el Mòdul de càmera V2 de Raspberry Pi
- Port de visualització DSI per connectar la pantalla tàctil de Raspberry Pi (no utilitzat)
- Port micro SD per connectar la targeta micro SD que conté el sistema operatiu i per emmagatzemar informació.
- Font d'alimentació micro USB 5V fins a 2.5A.

Dispositiu automàtic per a jugar a escacs Marçal Civill Cortada

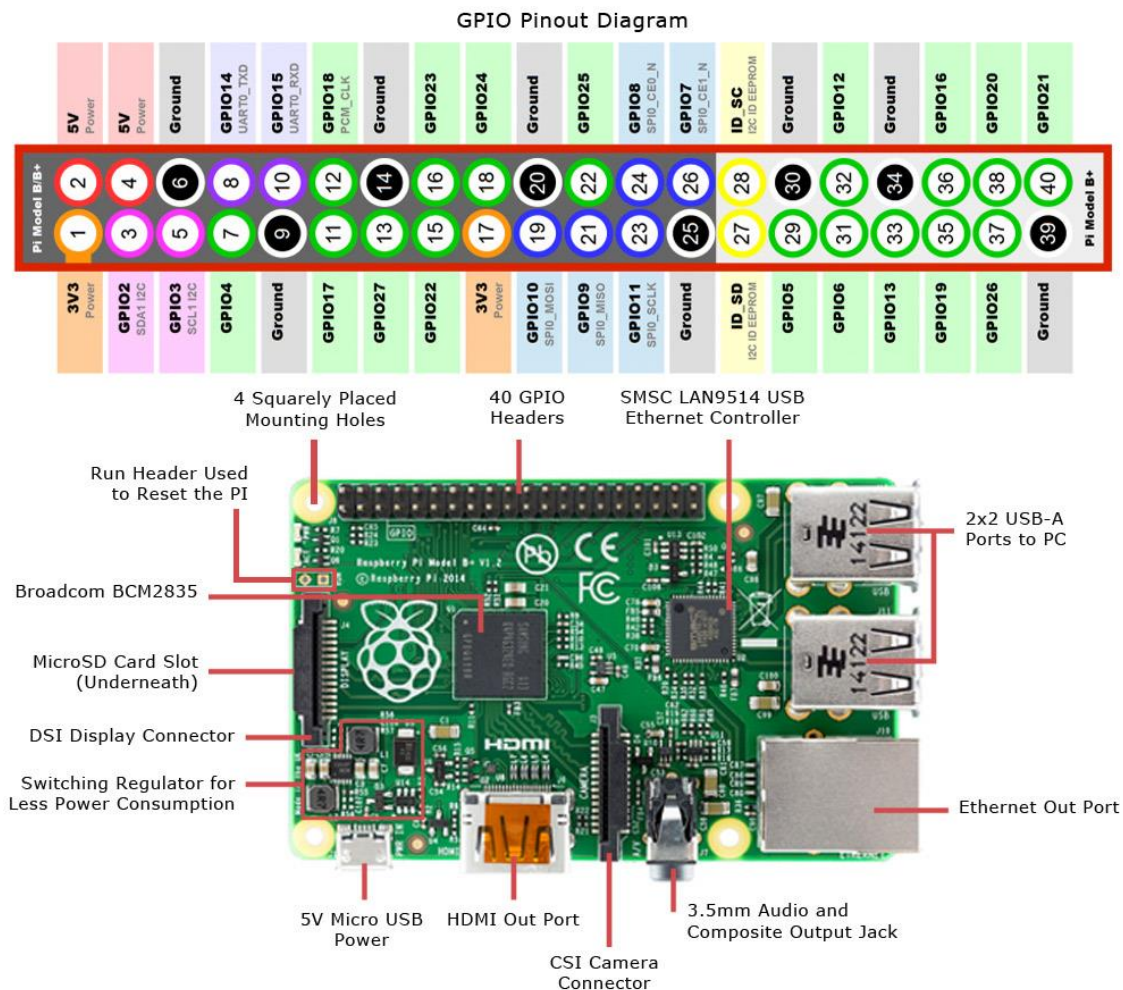


Figura 16. Esquema dels pins GPIO i altres connectors de la Raspberry Pi

2.1.8 SISTEMA OPERATIU

La Raspberry Pi és un petit ordinador que requereix d'un sistema operatiu per funcionar. Els sistemes operatius que suporta Raspberry Pi son distribucions de Linux per a arquitectura ARM. El recomanat pels fabricants de Raspberry Pi és Raspbian, ja que és el mes optimitzat i creat específicament per a ella, i es va decidir utilitzar-lo ja que no hi havia cap motiu aparent per utilitzar-ne un altre i complia tots els requisits.



Figura 17. Logo de Raspbian. (Font: <https://www.raspbian.org/>)

Raspbian és un sistema operatiu lliure derivat de Debian i optimitzat per al hardware de Raspberry Pi.

El sistema operatiu és el conjunt de programes i utilitats bàsics que fan que Raspberry Pi funcioni. Tanmateix, Raspbian proporciona més que un sistema operatiu pur, ve amb més de 35.000 paquets de programari pre-compilat i empaquetat en un format adequat per facilitar la instal·lació a Raspberry Pi. Entre aquests programes hi ha Python 3, el llenguatge de programació que s'ha utilitzat en aquest projecte.

Web oficial de Raspbian: <https://www.raspbian.org/FrontPage>

Documentació: <https://www.raspberrypi.org/documentation/raspbian/>

2.1.9 CÀMERA

Per a la captura d'imatges es va escollir el mòdul Raspberry Pi Camera Module V2. Un mòdul de càmera de 8-megapíxels dissenyat específicament per la Raspberry Pi.



Figura 18. Raspberry Pi Camera Module V2. (Font: <https://www.raspberrypi.org>)

El mòdul de la càmera és un petit PCB que es connecta al port de la càmera CSI-2 (Camera Serial Interface (versió 2)) de la Raspberry Pi mitjançant un cable de cinta. Proporciona connectivitat per a una càmera capaç de capturar imatges fixes o enregistraments de vídeo.

La càmera es vent normalment juntament a un cable de cinta curt, però per aquest projecte s'ha hagut de comprar un cable de cinta d'1m.

La càmera es connecta al sistema Image Pipe Line (ISP) en el SoC de Raspberry Pi, on es processen les dades de la càmera entrants i finalment es converteixen en una imatge guardada en la memòria (sigui la targeta SD o un altre emmagatzematge).

El mòdul de càmera V2 és un Sony IMX219, és comparable a les càmeres que s'utilitzen en telèfons mòbils.

Documentació:

<https://www.raspberrypi.org/documentation/hardware/camera/README.md>

2.1.10 CABLE DE LA CÀMERA

En aquesta màquina la càmera es situa en una estructura elevada obre el taulell, que l'he anomenat "estructura vertical". La càmera s'ha de connectar amb la Raspberry Pi situada per sota aquesta estructura. Per fer-ho el cable recorre l'interior de la estructura vertical fins a sortir per la base, travessant un forat tallat en la estructura principal. En aquest punt i just al costat del forat, es troba la Raspberry Pi i es connecta al seu port CSI (Camera Serial Interface). Aquest recorregut mesura poc menys d'un metre i el cable queda perfectament ajustat.

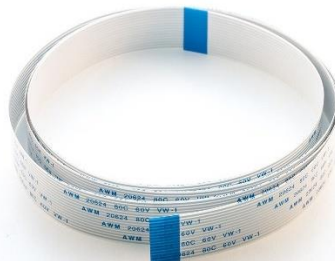


Figura 19. Cable d'1m per la càmera. (Font: <https://www.az-delivery.de/>)

2.1.11 TARGETA SD

La Raspberry Pi no disposa de memòria interna, està preparada per utilitzar com a memòria una targeta micro SD externa. En aquesta targeta es gravarà el sistema

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

operatiu i s'emmagatzemarà tota la informació a llarg plaç que s'introdueixi o es generi de nou dins de la Raspberry.

La capacitat necessària de la targeta és difícil de definir, depèn de l'ús que se'n vulgui fer i de la quantitat d'informació que hagi d'emmagatzemar. Normalment amb 8GB és suficient, però per no arriscar-me quan vaig començar el projecte vaig comprar una Micro SD de 16 GB.



Figura 20. Targeta MicroSD de 16 GB. (Font: <https://www.sandisk.es/>)

Documentació sobre la elecció de la targeta SD per a la Raspberry Pi:

<https://www.raspberrypi.org/documentation/installation/sd-cards.md>

2.1.12 FONT D'ALIMENTACIÓ 5V PER A LA RASPBERRY PI

La Raspberry Pi s'alimenta a través d'un port micro USB a 5V CC. La font que he utilitzat és una font estàndard de 5V 3A, que fan un total de 15W de potència màxima. Aquesta potència es considera perfectament suficient ja que el consum estimat de la Raspberry amb perifèrics connectats és d'entre 3,5W i 5W.



Figura 21. Font 5V 3A micro USB per Raspberry Pi. (Font: <https://www.amazon.es>)

Per fer-nos una idea del consum d'alguns perifèrics, els pins GPIO poden consumir 50 mA de manera segura, distribuïts en tots els pins; un pin GPIO individual només pot consumir amb seguretat 16 mA. El port HDMI utilitza 50 mA, el mòdul de la càmera requereix 250 mA i els teclats i ratolins poden gastar entre 100 mA o més de 1000 mA. S'ha de comprovar la potència dels dispositius que es volen connectar a la Raspberry i comprar una font d'alimentació en conseqüència.

També s'ha de tenir en compte que la màxima intensitat que pot utilitzar la Raspberry és 1A, s'ha de vigilar el consum dels dispositius que es connecten, i si consumeixen més s'han d'alimentar per fonts externes.

Documentació sobre l'alimentació de la Raspberry Pi:

<https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md>

2.1.13 TIPUS DE CONFIGURACIONS MECÀNIQUES RELLEVANTS

Hi ha molts tipus de configuracions mecàniques, aquestes són les més interessants i potencialment aplicables en aquest tipus de projecte:

Configuració cartesiana

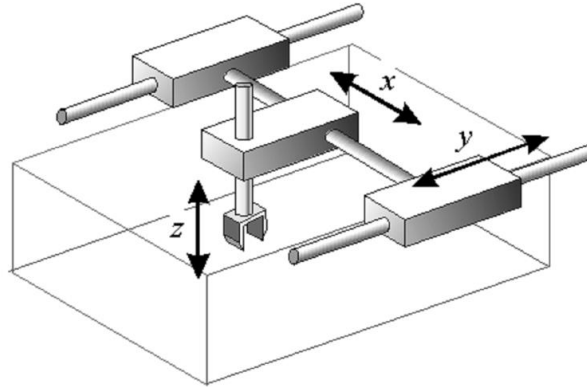


Figura 22. Esquema de la configuració cartesiana. (Font: <https://www.udesantiagovirtual.cl/wp/>)

En aquesta configuració, les tres articulacions són de tipus prismàtic. Les guies estan disposades de forma ortogonal entre elles en les direccions x, y i z, de manera que generen un espai d'actuació en forma de caixa rectangular.

Per la seva configuració, les coordenades de l'actuator són directament el recull idèntic de les coordenades de les articulacions. Per aquest motiu és un dels sistemes més senzills de controlar.

Configuració cilíndrica

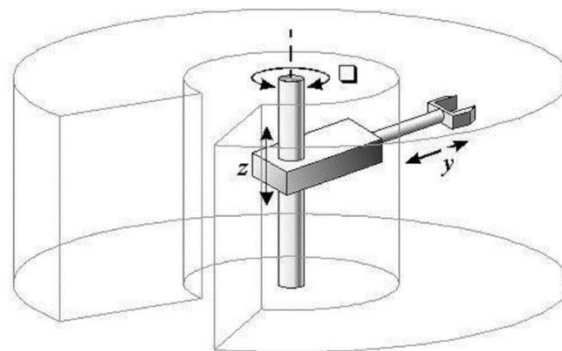


Figura 23. Esquema de la configuració cilíndrica. (Font: <https://www.udesantiagovirtual.cl/wp/>)

La primera articulació, la de la base, és de tipus cilíndric, el que genera una rotació en l'eix vertical. La segona i la tercera articulació són de tipus prismàtic.

Configuració rotacional

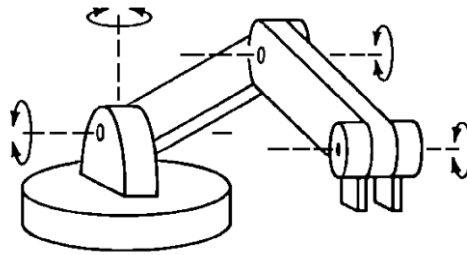


Figura 24. Esquema de la configuració rotacional. (Font: <https://www.udesantiagoovirtual.cl/wp/>)

Totes les articulacions són de tipus rotacional, totes alineades en el mateix eix horitzontal. També se l'anomena configuració antropomorfa per la seva similitud amb el braç humà.

Aquesta configuració ofereix un gran espai de treball, però el seu control és molt més complex que el de la configuració cartesiana, a causa de la major complexitat en l'anàlisi dinàmic.

Configuració SCARA

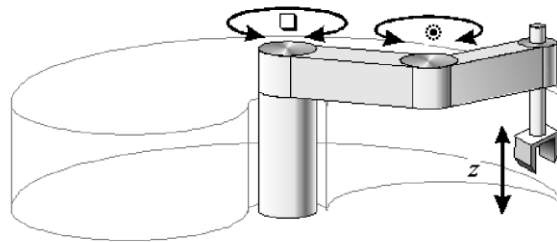


Figura 25. Esquema de la configuració SCARA. (Font: <https://www.udesantiagoovirtual.cl/wp/>)

SCARA "Selective Compliant Articulated Robot for Assembly" és una configuració on les dues primeres articulacions són de tipus rotacional, alineades en el mateix eix vertical. La tercera és una articulació prismàtica també orientada en l'eix vertical.

Configuració en paral·lel

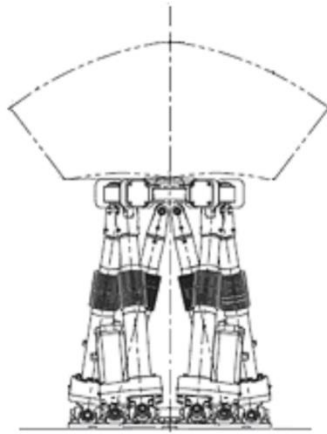


Figura 26. Esquema de la configuració en paral·lel. (Font: <https://www.udesantiagovirtual.cl/wp/>)

En aquesta configuració els esclavos es disposen en paral·lel en comptes de en sèrie. Les articulacions poden ser una combinació entre prismàtiques i rotacionals (exemple: robot hexàpode) o purament rotacionals (exemple: robot delta).

Aquesta configuració permet realitzar moviments de gran velocitat i acceleració a la vegada que els errors de posició no són acumulatius, a diferència de les configuracions en sèrie. El gran inconvenient és que l'espai de treball és molt reduït en comparació a les dimensions del sistema.

Web de referència (Campus virtual de la Universidad de Santiago de Chile):
<http://www.udesantiagovirtual.cl/moodle2/mod/book/view.php?id=24911>

2.1.14 CONFIGURACIÓ MECÀNICA ESCOLLIDA

Vaig escollir la configuració cartesiana ja que és la que millor s'adapta al tipus de muntatge que volia realitzar. La idea era poder moure les peces des de sota del taulell, per tant el sistema mecànic quedaria tancat en un espai en forma de caixa rectangular. Degut a això, la millor configuració era la cartesiana, ja que automàticament crea un espai d'actuació en forma de caixa rectangular estalviant així una gran quantitat d'espai.

Com que el taulell d'escacs és una superfície quadrada, el sistema cartesià també és el més eficient per recorre'l en termes d'abast del mecanisme.

Es poden distingir diverses sub-configuracions dins del grup de configuracions cartesianes segons la posició on es muntin els diversos components i el tipus de components que s'utilitzin.

La configuració típica és col·locar un motor fix sobre la bancada encarregat de moure la primera estructura mòbil. Sobre d'aquesta es col·loca un segon motor que es mou amb ella i s'encarrega de moure el segon patí, completant així un sistema capaç de moure's en 2D. El tercer motor, encarregat del moviment en Z, es posiciona sobre el segon patí completant així la configuració 3D. Però aquesta configuració suposa tres inconvenients importants:

- 1) El primer són els 2 motors movent-se juntament amb la estructura, això augmenta la massa que s'ha de moure, augmentant així la força i energia necessàries per al moviment.
- 2) El segon té a veure amb la energia elèctrica que s'envia als motors. Els cables que la transporten han d'estar en moviment. En un espai ampli això no suposa

cap problema, però dins una petita caixa, els cables en moviment poden comportar problemes de doblegament excessiu i repetitiu dels cables que pot ocasionar que es trenquin a causa de l'augment d'acritud o de la fatiga al cap d'un temps d'us prolongat.

- 3) Un altre problema que els cables en moviment poden ocasionar és el bloqueig d'altres peces mòbils, com per exemple enganxar-se en les corretges quan passen a través de les politges o col·locar-se sota les rodes quan es mouen sobre les guies, provocant així bloquejos de la màquina i en el pitjor dels casos també danys en els components.

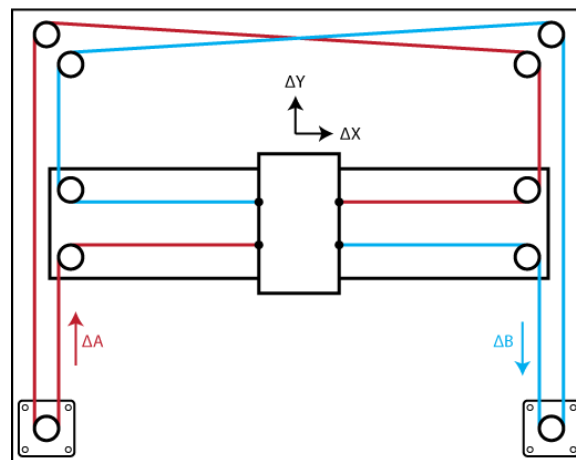
Per evitar aquests inconvenients es pot utilitzar la configuració CoreXY, que és la que he utilitzat en aquest projecte. Aquesta configuració manté 2 dels 3 motors estàtics, de manera que els inconvenients comentats anteriorment es redueixen considerablement, eliminant els 4 cables en moviment i la massa considerable d'un dels motors pas a pas.

core(X,Y)

Figura 27. Logotip de Core XY. (Font: <http://corexy.com/index.html>)

En aquest projecte, el tercer motor és un micro-servo, és extremadament petit i la seva massa és despreciable, però manté el problema de 3 cables movent-se en un espai molt limitat.

Aquesta configuració comporta una cinemàtica més complexa respecte la configuració cartesiana clàssica, ja que en la Core XY quan un motor es mou individualment l'actuador es mou en diagonal en comptes de moure's en línia recta.



Equations of Motion:

$$\Delta X = \frac{1}{2}(\Delta A + \Delta B), \quad \Delta Y = \frac{1}{2}(\Delta A - \Delta B)$$

$$\Delta A = \Delta X + \Delta Y, \quad \Delta B = \Delta X - \Delta Y$$

Figura 28. Esquema Core XY. (Font: <http://corexy.com/theory.html>)

Web oficial de Core XY: <http://corexy.com/index.html>

2.1.15 SISTEMA DE SUBJECCIÓ DE LES PECES

Per subjectar les peces es poden utilitzar diversos sistemes:

MECANIC

Una pinça de dos o tres dits és molt utilitzada però només es pot utilitzar per moure peces des de dalt del taulell. Com que en aquest projecte el mecanisme es mou per sota del taulell, una pinça no és aplicable.

ELECTRO-MAGNETIC

Aquest sistema es pot utilitzar tant des de dalt com des de sota del taulell.

- Si s'utilitza des de sobre del taulell, es requereix d'un sistema mecànic que a part de moure's en 2D també pugui moure l'imant en direcció vertical per acostar-lo i allunyar-lo de les peces. En aquest cas, l'electroimant entra en contacte directe amb una peça de ferro col·locada a la part superior de la peça. A causa d'aquest contacte directe, la força generada és molt gran i l'electroimant pot ser petit.
- Si s'utilitza des de sota del taulell, no és necessari un sistema mecànic que mogui l'actuador en 3D, un sistema 2D és suficient. Però presenta un gran inconvenient, degut a la distància que separa la peça de l'electroimant, ocasionada pel gruix del taulell, obliga a que l'electroimant sigui molt potent i en conseqüència de grans dimensions, un problema quan es vol reduir al màxim la mida de la màquina.

MECANIC + MAGNETIC PERMANENT

Aquest sistema només es pot utilitzar des de sota del taulell. Per aplicar aquest sistema, la base de les peces ha de contenir un imant o una peça de ferro que serà atreta per l'imant permanent de l'actuador.

L'imant permanent s'acosta a la base de la peça des de sota el taulell i les captura amb el camp magnètic. Seguidament, l'actuador es mou a la posició on es vol posicionar la peça, mantenint sempre l'imant a la mateixa distància del taulell amb l'objectiu de no perdre la força d'atracció amb la peça. Un cop arribat a la posició final, l'imant s'allunya mecànicament del taulell mogut per un motor, reduint així la força d'atracció entre la peça i l'imant permanent fins al punt en que és tècnicament inexistent. Quan l'imant de l'actuador està en aquesta posició, l'actuador es pot moure lliurement a la nova posició sense desplaçar cap peça en el procés.

2.2 RECERCA POSTERIOR AL PRE-DISENY

2.2.1 MOTORS PAS A PAS

El motor escollit és el Motor pas a pas NEMA 17 1.7A Model: 17HS4401. Un motor molt utilitzat en impressores 3D i fresadores de petit format. És un motor pas a pas amb $1,8^\circ$ de resolució, un parell de retenció de 2,2N·cm i una intensitat màxima de 1,7A.



Figura 29. Foto del motor NEMA 17. (Font: <http://www.motionking.com>)

NEMA 17 és un format de motor pas a pas estandarditzat per NEMA (National Electrical Manufacturers Association) amb unes mides entre cares laterals de 1,7 x 1,7 polzades (42,3 x 42,3 mm).

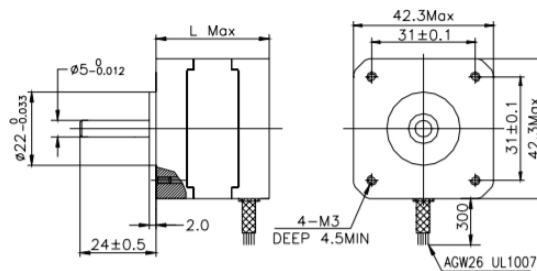


Figura 30. Esquema motor NEMA 17. (Font: <http://www.motionking.com>)

Datasheet: http://www.motionking.com/download/HB_Stepper_Motor_E.pdf

2.2.2 DRIVERS PER ALS MOTORS

El control dels motors pas a pas no es pot fer directament des de la Raspberry Pi, ja que no pot subministrar la potencia necessària per alimentar-los (en aquest cas és una Raspberry Pi però passaria el mateix amb qualsevol altre microcontrolador).

Per poder controlar i alimentar els motors, entre la Raspberry i el motor es col·loca un driver. El driver rep per un costat les senyals de control del microcontrolador a la vegada que rep l'alimentació per part d'una font externa, en aquest cas una font de 12V i 3A. Per altra banda el driver envia l'alimentació al motor a través de quatre cables. Cada bobina o conjunt de bobines del motor correspon a un parell de cables que seran alimentades alhora en la sincronització exacta perquè el motor giri a la velocitat i direcció que se l'hi indiqui des del microcontrolador.

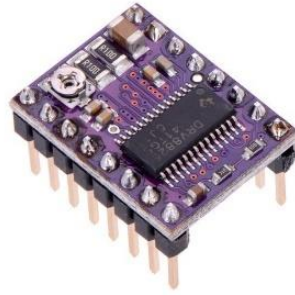


Figura 31. Foto del driver DRV8825. (Font: <https://www.pololu.com>)

Per controlar els motors NEMA 17 comentats anteriorment s'ha escollit el driver DRV8825, un driver molt comú utilitzat en impressores 3D i màquines CNC, molt econòmic i que resisteix una elevada potencia.

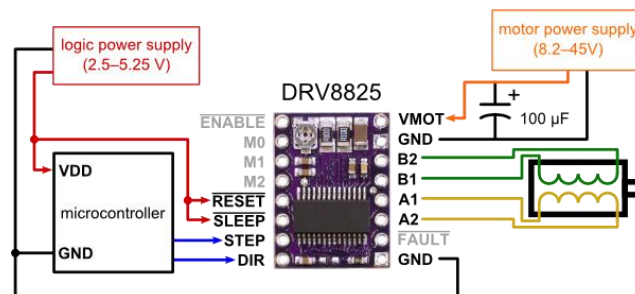


Figura 32. Esquema de connexions del driver DRV8825. (Font: <https://www.pololu.com>)

Datasheet: <https://www.pololu.com/file/0J590/drv8825.pdf>

2.2.3 FONT D'ALIMENTACIÓ 12V

Els motors pas a pas no es poden alimentar directament de la Raspberry pi, ja que aquesta no pot gestionar tanta potencia, per això s'utilitza una font externa gestionada per la Raspberry a través dels drivers comentats anteriorment.

Els drivers es poden alimentar amb un voltatge d'entre 8,2V i 45V CC. La font que he escollit entre aquest marge és una font de 12V i 3A, una font molt comuna i per això força econòmica i fàcil de trobar en qualsevol botiga d'electrònica.



Figura 33. Font d'alimentació 12V 3A. (Font: <https://www.amazon.es>)

2.2.4 CONNECTOR JACK-REGLETA PER A FONT D'ALIMENTACIÓ

La font d'alimentació de 12V comentada anteriorment té una sortida Jack 2,5mm mascle, per poder connectar la font als diversos components tenia tres opcions:

- Soldar una placa amb un connector Jack de 2,5mm femella i en paral·lel soldar els diversos pins o regletes de sortida on connectar els cables cap als diversos components.
- Tallar el cable i retirar el connector Jack, separar els cables, pelar-los i connectar-los a una regleta o soldar-los en una placa amb regletes soldades en paral·lel.
- La última i la que vaig trobar més ràpida i més interessant ja que estalviava feina i em permetia conservar la font d'alimentació intacta per futurs projectes era utilitzar un connector Jack 2,5mm femella – regleta com es pot veure en la fotografia:



Figura 34. Connector jack femella a regleta per la font d'alimentació de 12V. (Font: <https://www.amazon.es>)

2.2.5 SERVOMOTOR

Un servomotor molt econòmic és el micro-servo SG90 molt utilitzat en projectes que requereixen poca potència en els moviments com es el cas d'acostar i retirar l'imant que atrau les peces des de sota el taulell.



Figura 35. Motor SG90 previst d'utilitzar en el sistema de subjecció magnètica de les peces. (Font: <http://tienda.bricogeek.com>)

Datasheet: <http://akizukidenshi.com/download/ds/towerpro/SG90.pdf>

2.2.6 VENTILADORS

Per a la refrigeració de l'interior de la caixa s'ha instal·lat dos ventiladors de 50x50 mm en dues de les parets laterals a l'altura del muntatge electrònic.



Figura 36. Ventilador 12V 50x50mm. (Font: <https://www.amazon.es>)

Un ventilador absorbeix aire de l'exterior i l'introdueix a la caixa a la vegada que l'altre expulsa l'aire calent a l'exterior.

Aquests ventiladors només disposen dels dos cables d'alimentació i estan connectats directament a la font de 12V que els alimenta, així que quan s'engega la màquina, els ventiladors s'engeguen automàticament.

2.2.7 POTENCIÒMETRES ENCODERS

La posició d'un actuador mogut per motors pas a pas es pot controlar de dues maneres diferents:

1. **LA PRIMERA** implica que al engegar la màquina, els motors han de moure l'actuador en direcció a la posició de calibració fins que entri en contacte amb els diversos sensors de final de carrera que indicaran que ha arribat a la posició inicial. Un cop activats tots els sensors de final de carrera l'actuador es troba en la posició (0,0,0). A partir d'aquí, en cada moviment que faci la màquina, el microcontrolador ha de portar el compte dels passos que ha indicat fer a cada motor pas a pas. El programa anirà sumant passos quan el motor giri allunyant l'actuador del punt de calibració i restant-los quan s'hi acosti.

Aquest sistema ofereix un avantatge i diversos desavantatges:

a. Avantatges:

- No requereix un sistema d'encoders per controlar la posició de l'actuador

b. Desavantatges:

- El microcontrolador només pot portar el compte dels passos que ha indicat fer al motor pas a pas, però no pot controlar els passos reals que aquest ha fet. Així, si algun dels motors perd un o més passos en algun moviment on l'actuador s'hagi encallat, el microcontrolador no tindrà manera de saber-ho, i per tant suposarà que l'actuador està en una posició que no és la real.
- El segon problema que suposa aquest sistema és que quan la màquina no ha de realitzar cap moviment i està parada, els motors pas a pas han de mantenir la posició. Per mantenir la posició el motor pas a pas ha de mantenir les bobines electrificades i encara que sembli contra intuïtiu, quan està en aquesta situació, aturat mantenint la posició, és quan gasta més energia. Això no és un problema en una impressora 3D o

fresadora, ja que la màquina està en continu moviment des de que comença a treballar fins que acaba. Però la màquina d'aquest projecte requerirà estar parada molta estona esperant la jugada de l'humà i això suposaria un consum elèctric molt elevat.

- El tercer inconvenient deriva del segon, si els motors estan aturats durant una llarga estona consumint una quantitat considerable de potència, això provocarà un escalfament elevat i prolongat en el temps tant dels motors com dels drivers que generaran una elevada quantitat de calor per efecte Joule. En la configuració d'aquesta màquina, els motors i especialment els drivers, estan tancats en un espai reduït juntament amb altres components electrònics que requereixen un rang de temperatura ambient adequat per treballar i tenir una adequada refrigeració. Si la temperatura de l'interior de la caixa s'eleva excessivament seran necessaris sistemes específics de refrigeració per reduir la temperatura.

2. **LA SEGONA** opció implica la instal·lació d'un sistema d'encoders que mesurin contínuament la posició de l'actuador. Hi ha dos tipus d'encoders:

- a. **ENCODER RELATIU:** Mesura el moviment però no la posició. Quan el motor gira emet polsos elèctrics cada vegada que el motor gira una certa quantitat de graus i el microcontrolador pot anar contant aquests impulsos de manera que pot extrapolar la posició de l'actuador en tot moment.

Aquest sistema presenta diversos avantatges i desavantatges:

Avantatges:

- No tenen limitació de rotació, poden girar infinitament en una direcció o en la contrària sense límit de voltes. Això és útil quan el motor ha de ser capaç de girar més de 10 voltes en la mateixa direcció. Per sobre de 10 voltes és molt difícil trobar encoders absoluts.

Desavantatges:

- Aquest sistema presenta l'inconvenient d'haver de portar l'actuador a la posició inicial cada vegada que s'engega la màquina. Per tant també requereix la instal·lació de finals de carrera i un fragment de programa específic per a la cal·libració.
- També necessita un programa específic que conti tots els impulsos que emet l'encoder en temps real. Es pot utilitzar un microcontrolador entre el microprocessador i l'encoder, però això suposa ampliar el muntatge electrònic.

- b. **ENCODER ABSOLUT:** Aquest tipus d'encoders emeten un valor analògic contínuament que varia amb la posició de l'encoder. En qualsevol moment es pot consultar aquest valor i calcular la posició de l'actuador amb precisió.

Avantatges:

- No requereixen software específic dedicat i es pot conèixer la posició exacta de l'actuador en qualsevol moment i sense cal·libració inicial.

Inconvenients:

- Al ser dispositius mecànics, com ara potenciómetres, pateixen

un desgast durant el seu ús i tenen una vida útil limitada.

- Sent dispositius mecànics que emeten un valor de posició absolut, requereixen tenir un límit físic inicial i un límit final. Això provoca que aquests encoders puguin mesurar un rang de posicions limitat.

Els encoders utilitzats en aquest projecte són dos potenciòmetres BOURNS 3590S-2-103 amb un rang de 10 voltes i una resistència de 10k ohms.



Figura 37. Foto del potenciòmetre BOURNS multivolta. (Font: <https://www.electronicaembajadores.com>)

La quantitat màxima de voltes que ha de fer cada un dels motors per moure l'actuador entre la seva posició mínima i la màxima és d'aproximadament 8 voltes. L'eix del potenciòmetre s'engranarà amb l'eix del motor mitjançant dues rodes dentades amb una relació de transmissió de 1:1. Els detalls sobre els engranatges es dona en el següent apartat.

Datasheet: <https://www.bourns.com/pdfs/3590.pdf>

2.2.8 ENGRANATGES PER AL SISTEMA D'ENCODERS

Per a connectar mecànicament el motor i el potenciòmetre que fa la funció d'encoder, es va decidir utilitzar engranatges a 45° entre els dos eixos.

Són engranatges a 45° RS Pro de material POM, 20 dents i calibre 5 mm.



Figura 38. Foto de l'engrenatge a 45° de RS Pro. (Font: <https://es.rs-online.com>)

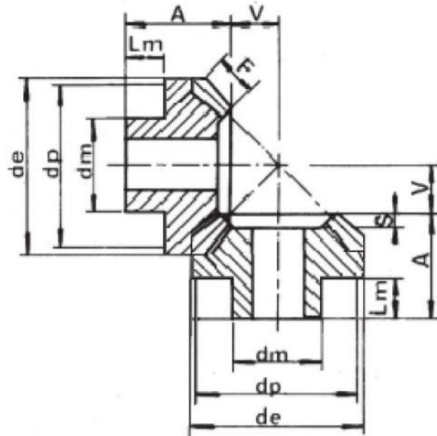


Figura 39. Muntatge relatiu entre les dues rodes dentades. (Font: <https://docs-emea.rs-online.com/webdocs/157a/0900766b8157a694.pdf>)

Es muntaran dues rodes dentades iguals tant a l'eix del motor com a l'eix del potenciòmetre, el que produirà una relació de transmissió de 1:1, ja que el motor requereix d'aproximadament 8 voltes per portar l'actuator d'un vèrtex del taulell al vèrtex oposat, el recorregut més llarg possible. El potenciòmetre per la seva banda té un límit de 10 voltes, per tant, una relació de 1:1 és la més adequada.

Els dos eixos s'hauran de muntar en la posició que es mostra a l'esquema anterior.

Aquests engranatges, per anar be, requereixen un muntatge amb bona precisió i bon ajust. Com que aquesta màquina l'estic fabricant sense maquinaria de precisió tipus CNC, l'ajust s'ha fet manualment i encara que no és l'ajust ideal, s'adapten prou be i gairebé no oposen resistència al girar.

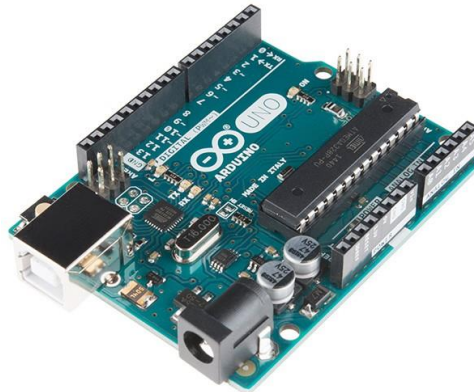
Datasheet: <https://docs-emea.rs-online.com/webdocs/157a/0900766b8157a694.pdf>

2.2.9 CONVERSOR AD PER A LA LECTURA D'ENCODERS

Al utilitzar dos potenciòmetres com a encoders absoluts que emeten una senyal analògica apareix un problema. La Raspberry Pi no és capaç de llegir les senyals analògiques i per tant requereix d'una transformació de la senyal d'analògica a digital (AD) entre els potenciòmetres i la Raspberry.

Eren varies les solucions que es podien adoptar aquí, però les més senzilles eren utilitzar un microcontrolador entre els potenciòmetres i la Raspberry Pi.

Inicialment vaig utilitzar un Arduino UNO, una placa basada en el microcontrolador ATmega328. La vaig utilitzar perquè el tenia a mà i ja l'havia utilitzat en alguna altra ocasió. El que vaig fer va ser llegir els potenciòmetres amb l'Arduino i comunicar l'Arduino a la Raspberry mitjançant una simulació de connexió sèrie a través d'un port USB.



Quan l'hi vaig comentar al tutor em va recomanar utilitzar una solució similar però utilitzant un PIC en comptes d'un Arduino, cosa que reduïa la electrònica considerablement.

La solució adoptada va ser llegir la senyal analògica dels potenciòmetres mitjançant el microcontrolador PIC 12F1822. Els dos potenciòmetres estan connectats a dues de les potes d'entrada analògica del PIC i aquest a la vegada està connectat a la Raspberry Pi mitjançant els dos ports GPIO de comunicació sèrie.



Figura 40. Foto del PIC 12F1822. (Font: <http://www.microchip.com>)

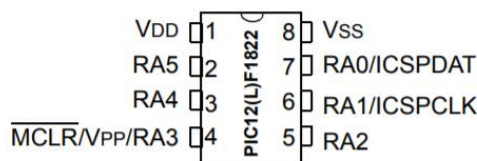


Figura 41. Diagrama de pins del PIC12F1822. (Font: <http://ww1.microchip.com/downloads/en/DeviceDoc/40001413E.pdf>)

Datasheet: <http://ww1.microchip.com/downloads/en/DeviceDoc/40001413E.pdf>

2.2.10 POLITGES I CORRETGES

La idea inicial era utilitzar un vis sense fi per transmetre el moviment, però es va comprovar que els visos sense fi de la longitud requerida eren molt cars, per això es va plantejar utilitzar varilla roscada en comptes de vis sense fi.

Aquí va aparèixer un gran problema, la varilla roscada de les dimensions adequades tenia un pas del voltant d'1mm, això significa que la varilla havia de girar a una velocitat extremadament elevada per moure les peces d'escacs a un ritme en que el jugador no s'avorrís d'esperar. La velocitat que es plantejava era extrema, utilitzar aquest sistema hagués suposat unes vibracions i soroll molt elevats a més de generar unes grans inèrcies, de manera que si la varilla no era perfectament recta, que seria d'esperar, començaria de costat a costat amb gran violència.



Figura 43. Vis sense fi per impressora 3D.
(Font: <https://www.amazon.es>)

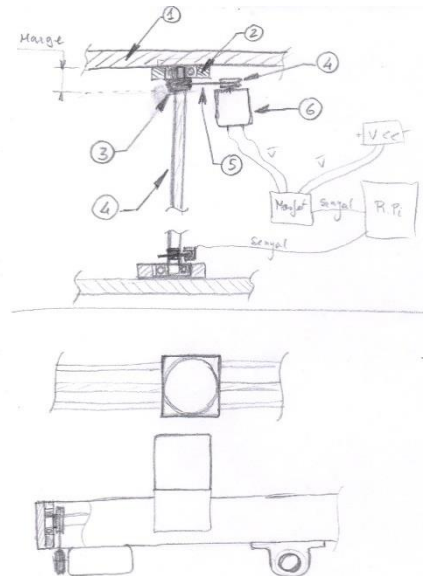


Figura 43. Esquema de muntatge del sistema de transmissió amb vis sense fi. (Font: Elaboració pròpia)

Sumant-se a totes aquestes consideracions també estava el fet que accelerar la varilla a aquestes velocitats suposava un cert temps i molta energia i per frenar s'hagués hagut d'instal·lar algun tipus de fre mecànic.

Fets aquests plantejaments la idea del vis sense fi es va descartar i es va buscar un altre sistema. Buscant es va trobar que un sistema molt utilitzat en impressores 3D, a part del vis sense fi, eren les corretges. Normalment el vis sense fi s'utilitzava per impressores petites d'alta precisió o en impressores 3D comunes per transmetre el moviment vertical, ja que aquest és molt lent. Per al moviment en pla en molts models s'utilitzaven sistemes de politges i corretges que permetien transmetre molta més velocitat..

Les politges i corretges utilitzades en aquest projecte són del model GT2, un model compatible amb moltes de les impressores 3D lliures disponibles en el mercat que formen part del projecte Rep Rap un projecte Open Source orientat a les impressores 3D. Això fa que sigui molt fàcil comprar-les a molt baix cost a qualsevol botiga d'impressores 3D o per internet.



Figura 44. Logo de RepRap. (Font: <http://reprap.org>)

El diàmetre del forat central en el cas de les politges és de 5mm que correspon al diàmetre de l'eix dels motors NEMA, uns motors estandarditzats i també molt utilitzats en impressores 3D com he comentat anteriorment.

2.2.11 POLITGES MOTORS

Inicialment les politges que pretenia fer servir eren les que s'utilitzen més habitualment, les de 20 dents ja que són les més barates.



Figura 45. Politges GT2 de 20 dents. (Font: <https://www.amazon.es>)

Però després de fer proves de velocitat amb un motor pas a pas vaig comprovar que la velocitat amb que es mourien les peces era massa lenta per a un joc d'escacs, ja que esperar tant de temps a que la màquina fes el moviment es faria pesat i avorrit. Per això vaig buscar unes politges de més diàmetre per així augmentar la velocitat lineal amb la mateixa velocitat angular limitada pel motor.

També es podria utilitzar un sistema d'engranatges multiplicadors però si es podia trobar una politja amb el diàmetre suficient i m'estalviava el sistema d'engranatges el disseny es simplificaria considerablement.

Finalment les vaig trobar, politges de 60 dents les més grans que existeixen del model GT2. Respecte a les anteriors de 20 dents suposaven un augment de 3 vegades el perímetre, per tant la velocitat lineal de la corretja es multiplicaria per 3. Amb aquestes noves politges la velocitat que trigaria una peça a recórrer la diagonal del taulell seria inferior a 2 segons, una velocitat prou acceptable.



Figura 46. Foto de les politges motors. (Font: <https://www.amazon.es>)

El material d'aquestes politges és d'alumini, igual que el de la resta de politges utilitzades, això fa que el seu moment d'inèrcia sigui molt reduït aconseguint així reduir el parell necessari generat pel motor a l'ora d'accelerar.

2.2.12 POLITGES LLIURES DENTADES

El sistema mecànic utilitzat en aquest projecte, la configuració cartesiana CoreXY és un elaborat mecanisme de politges i corretges que permet mantenir fixes els dos motors principals.

Les úniques politges que generen tracció són les dues politges comentades anteriorment que s'instal·len en l'eix dels motors. La resta són politges lliures, també anomenades politges boges que giren lliurement mogudes per la corretja i tenen la funció de desviar la corretja en la direcció desitjada.

Segons si la corretja passa per la politja apoiant-se sobre la seva cara dentada o la seva cara plana s'han d'utilitzar politges lliures dentades o planes respectivament.

Fer recórrer la cara dentada de la corretja la superfície d'una politja plana pot suposar un augment en el desgast de les dents, a la vegada que el desgast es produeix amb més irregularitat cosa que pot provocar un mal funcionament a llarg plaç de la màquina i una degradació prematura de la corretja. Per això és important utilitzar els dos tipus de politges, tant les planes com les dentades quan correspongui per evitar problemes.



Figura 47. Politges lliures dentades GT2. (Font: <https://www.amazon.es>)

En aquest muntatge s'utilitzen 6 politges lliures dentades, dues d'elles es munten en els dos extrems de la guia fixe i les quatre restants es munten per parelles a les dues cantonades de la caixa oposades als motors.

Aquestes politges estan fetes d'alumini i porten incorporat un rodament central que permet que girin lliurement sense fricció.

2.2.13 POLITGES LLIURES LLISES

Com he comentat anteriorment, les politges planes s'utilitzen per guiar la cara plana de la corretja.



Figura 48. Politges lliures llises GT2. (Font: <https://www.amazon.es>)

En aquesta màquina se n'utilitzen dues que van muntades sobre la guia mòbil i desvien la corretja des del motor a l'actuador.

Aquestes politges, igual que les lliures dentades, estan fetes d'alumini i porten incorporat un rodament central que permet que girin lliurement sense fricció.

2.2.14 CORRETJA

La corretja utilitzada es del tipus GT2 oberta, medeix 6mm d'ample i esta feta de goma reforçada amb fibra.



Figura 49. Corretja GT2. (Font: <https://www.3dpenedes.com>)

És una corretja petita i molt flexible perfecte per a muntatges de petites dimensions.
En aquest projecte se'n necessiten don trams d'aproximadament 2,5 metres cada un.

2.2.15 TENSOR DE CORRETJA

Un cop muntada la corretja en la seva posició, pot ser que no quedi perfectament tensada, ja que normalment la corretja es lliga en el suport plegant-se sobre si mateixa i unint les seves pròpies dents al voltant d'un eix central de suport.

Això fa que la resolució amb la que es pot tensar inicialment la corretja sigui igual a la distancia entre dents de la corretja, el pas. Per acabar d'ajustar la mida i evitar que la corretja quedi destensada s'utilitzen tensors. Uns tensors de la mida adequada per a la corretja GT2 son aquests:



Figura 50. Tensor de corretja. (Font: <https://www.3dpenedes.com>)

Aquests tensors es col·loquen en els extrems de la corretja i s'ha de dissenyar la màquina tenint-los en compte, ja que el tram de corretja ocupat pel tensor no podrà passar a través d'una politja.

2.2.16 RODES

Inicialment va ser complicat decidir el tipus de rodes que s'utilitzarien, es van fer diversos esquemes i plantejaments respecte al tipus de roda per a diferents tipus de configuracions, des de rodes quadrades que s'apoiarien a les parets de la caixa, passant per rodes acabades en punta a menys de 90° per moure's per les arestes interiors de la caixa, fins a rodes amb canal rodó.

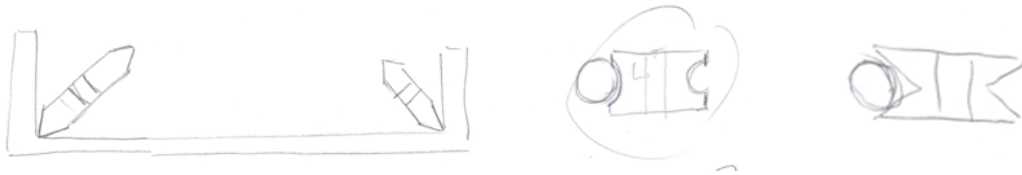


Figura 51. Esquemes de diferents tipus de rodes. (Font: Elaboració pròpia)

Finalment vaig descobrir les rodes V-Slot i vaig considerar que per mida, forma i preu eren les més adequades.



Figura 52. Rodes tipus V-Slot. (Font: <https://www.amazon.es>)

2.2.17 GUIES COMERCIALS

Hi ha molts tipus de guies al mercat amb un rang de preus ampli i en general elevat, utilitzar una guia comercial és molt car i normalment les més assequibles són massa curtes per a l'aplicació que requerien en aquest projecte.



Figura 53. Tres tipus diferents de guies lineals. (Font: <https://www.amazon.es>)

Altres tipus de guies específiques per al tipus de roda que volia utilitzar també eren cares i a la vegada ocupaven molt de volum.



Figura 54. Guia per a roda V-Slot. (<http://openbuildspartstore.com>)

Així que es va dissenyar un mètode de guia alternatiu per abaratir costos. Aquest està explicat en l'apartat de disseny.

3. DISSENY I FABRICACIÓ

3.1 DISSENY DEL TAULELL

En els inicis del projecte es pretenia utilitzar el taulell que es veu a la foto inferior, un taulell de fusta de molt bona estètica juntament amb unes peces de fusta a joc.

Aquest plantejament es va acabar descartant en part per els problemes que es van trobar durant les proves de subjecció magnètica:

- Les peces eren massa pesades per ser mogudes per un imant des de sota del taulell.
- El taulell era massa gruixut (10 mm aprox.) per permetre una atracció magnètica suficient entre l'imant i de l'actuador mecànic que s'havia de moure per sota el taulell i l'imant de la peça.



Figura 55. Taulell i peces que es preveien utilitzar inicialment. (Font: Elaboració pròpia)

Als problemes amb la subjecció magnètica s'hi van sumar els problemes amb l'anàlisi d'imatge:

- Les línies i formes de les vetes de la fusta tant del taulell com de les peces provocaven molts falsos positius quan es feia la detecció de cercles de la imatge per detectar les bases de les peces.
- El contrast de color entre les peces blanques i els escacs negres era molt petit i feia difícil la detecció de les peces si no hi havia una il·luminació perfecta. Passava que les peces de la perifèria no estaven tan ben il·luminades com les del centre i a més la perspectiva feia que les seves bases no es veiessin perfectament circulars, sinó més aviat el·líptiques. Si a això se l'hi sumava la falta de contrast en el color la detecció es feia pràcticament impossible.



Figura 56. Detall de les peces i el taulell inicials on s'observen les vetes de la fusta i el poc contrast de color entre els escacs negres i les peces blanques. (Font: Elaboració pròpia)

Descartat l'ús del taulell comentat anteriorment, el taulell va passar per dues etapes de disseny principals, inicialment es preveia fer servir una fusta prima per cobrir tota la caixa i que a la vegada faria de taulell, això feia que les seves dimensions fossin molt grans i podia provocar que es corbés sota el seu propi pes i el de les peces.

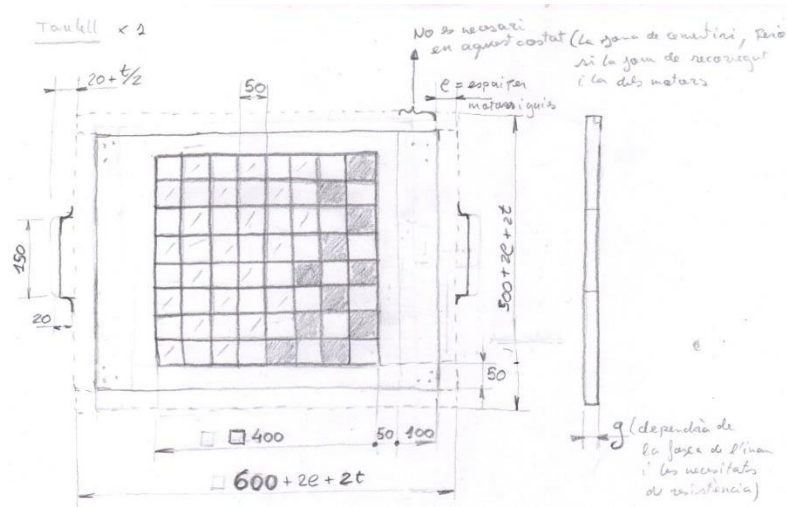


Figura 57. Esquema inicial del taulell. (Font: Elaboració pròpia)

Finalment aquest disseny es va descartar en favor d'una estructura principal gruixuda on es collarien la resta de peces i que aportaria resistència estructural al conjunt, inclòs el propi taulell, ja que aquest havia de ser prim perquè la atracció magnètica entre l'actuador (que es mou per sota) i les peces (col·locades a sobre) fos prou forta per moure-les.

El disseny físic final del taulell és una fusta contraxapada rectangular de 5mm de gruix i unes dimensions laterals de 460mm x 550mm.

Respecte al disseny gràfic es va apostar per una combinació de colors llisos i clars que contrastessin bé amb les peces de color negre i les de color beix o crema, que són els colors més utilitzats en les peces d'escacs.

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

Aquest disseny s'imprimiria en un vinil adhesiu i s'enganxaria sobre el taulell de fusta contraxapada de 5mm tallat prèviament.

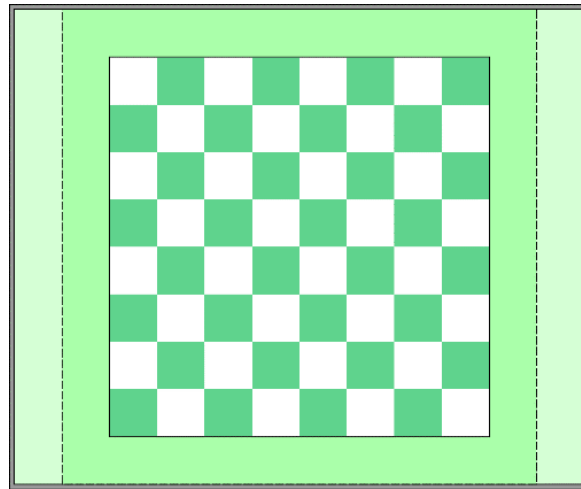


Figura 58. Imatge del taulell d'escacs. (Font: Elaboració pròpia)

El taulell es compon d'una matriu central de 8x8 escacs, un camí que rodeja aquesta matriu i dues zones de cementiri als laterals.

Els escacs medeixen 45x45mm, el camí medeix 45mm d'ample i les zones de cementiri també.

La matriu central correspon al taulell d'escacs tradicional.

El camí lateral compleix la funció de crear una zona de pas lliure perquè la màquina pugui moure les peces amb més facilitat des de sota del taulell.

Les zones cementiri són espais on la màquina deixarà les peces mortes del seu rival (el jugador humà) quan les retiri del taulell.

Finalment la zona grisa que rodeja el taulell medeix 5mm i és la zona on el taulell de fusta s'apoiarà sobre els llistons de la estructura principal.



Figura 59. Taulell de fusta contraxapada de 5mm amb el vinil enganxat. Al fons forat en l'estructura principal amb encaix per al taulell. (Font: Elaboració pròpia)

3.2 LA ELECCIÓ DE LES PECES

La idea inicial era utilitzar unes peces de fusta força grans per la seva bona estètica i la bona sensació al tacte. Més endavant i després de fer proves amb la subjecció magnètica es va veure que pesaven massa i s'havien de canviar per unes altres.

Es van escollir les que es veuen en la foto següent:



Figura 60. Vista detallada de les peces d'escacs. (Font: Elaboració pròpia)

Són unes peces més petites i molt més lleugeres que les inicials de fusta massissa, ja que aquestes eren de plàstic i amb l'interior buit.



Figura 61. Comparativa entre les peces que es preveien utilitzar inicialment i les que s'han acabat utilitzant. (Font: Elaboració pròpia)

Es va tenir en compte que la reducció de mida de les peces no acabés provocant una mala sensació en el joc, ja que la gràcia d'aquest taulell d'escacs és la interacció física i si aquesta no era bona perdria el sentit. Per aquest motiu es va voler mantenir una mida mínima dels escacs de 45mm i una mida mínima de les peces les bases de les quals varien entre 22 mm i 28 mm entre els peons i el rei respectivament. Això també estableix una bona relació de proporcions entre els escacs i les peces agradable a la vista.



Figura 62. Comparativ entre la proporció d'escac ocupat per les peces previstes inicialment i les finals. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

A més aquestes peces presenten una peculiaritat i és que la seva base és desmuntable i s'hi pot instal·lar fàcilment un imant a la cara interior. En aquest cas s'ha utilitzat cinta adhesiva de doble cara. Aquesta característica és molt útil per al muntatge que s'ha de realitzar en aquest projecte.



Figura 63. Peces amb la tapa inferior posada i treta on es pot veure el muntatge de l'imatge permanent interior. (Font: Elaboració pròpia)

Finalment les peces col·locades sobre el taulell presenten un bon aspecte i compleixen totes les característiques tècniques mecàniques i d'anàlisi d'imatge necessàries.



Figura 64. Vista del taulell amb les peces col·locades. (Font: Elaboració pròpia)

3.3 DISSENY DEL SISTEMA MECÀNIC I ESTRUCTURAL

L'objectiu ha estat realitzar un disseny mecànic el més simple i econòmic possible a la vegada que robust i elegant.

Per reduir el cost s'han evitat utilitzar guies i estructures comercials, ja que són unes de les peces més cares de les màquines que es troben al mercat. Per contra s'han dissenyat i desenvolupat dissenys nous que fabricats industrialment serien extremadament barats.



Figura 65. La màquina vista des de sota sense la tapa inferior. (Font: Elaboració pròpia)



Figura 66. Vista general de la estructura. (Font: Elaboració pròpia)

3.3.1 PROCÉS DE DISSENY. LA CAIXA.

La idea de la caixa va passar per diverses fases de disseny, aquí es pot veure un disseny a base de diverses parts que s'encaixen entre elles.

Més endavant quan es va decidir que les guies fixes s'assentarien sobre les parets laterals aquest format de caixa es va descartar ja que no permetia ajustar la seva posició.

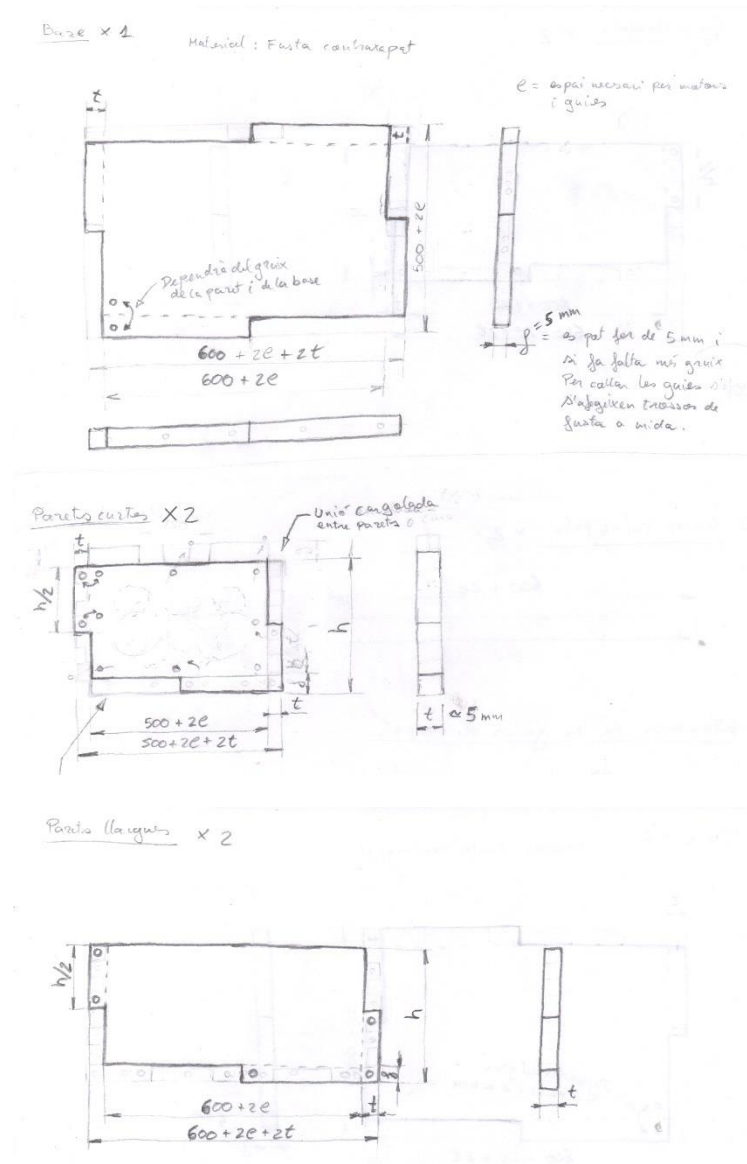


Figura 67. Pre-disseny de la caixa. Caixa puzle. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

Més endavant es va desenvolupar la idea de fer una caixa a partir de tubs rectangulars d'alumini units entre ells amb escaires rematxats o unions desmuntables a base d'inserts en escaire.

Posteriorment també es va descartar per la dificultat en el muntatge tant de la pròpia estructura com de la resta de peces que s'hi ha d'unir. També va influir la impossibilitat d'ajustar-la.

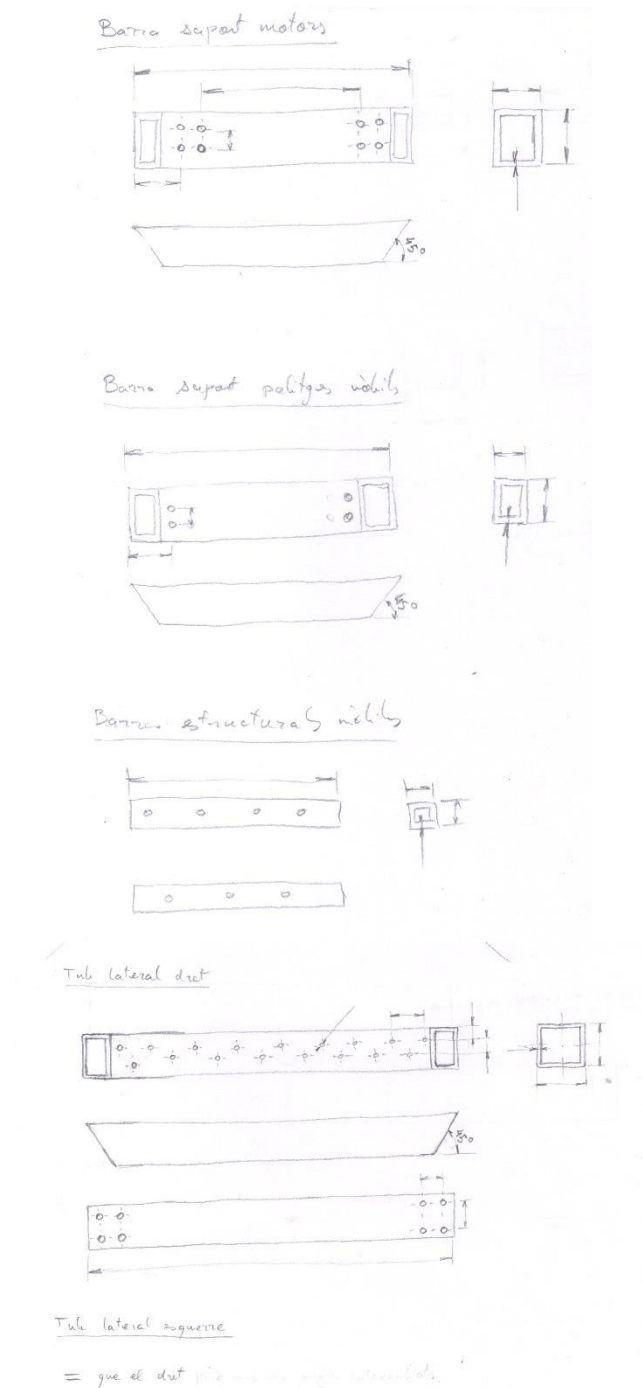


Figura 68. Pre-disseny de la caixa. Caixa amb parets de tub. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

Finalment es va optar per al disseny d'una caixa la tapa superior de la qual faria d'estructura principal a la qual es collarien la resta d'elements. Entre aquests elements es contenen les parets laterals unides a la estructura principal a través d'escaires. Aquests escaires.

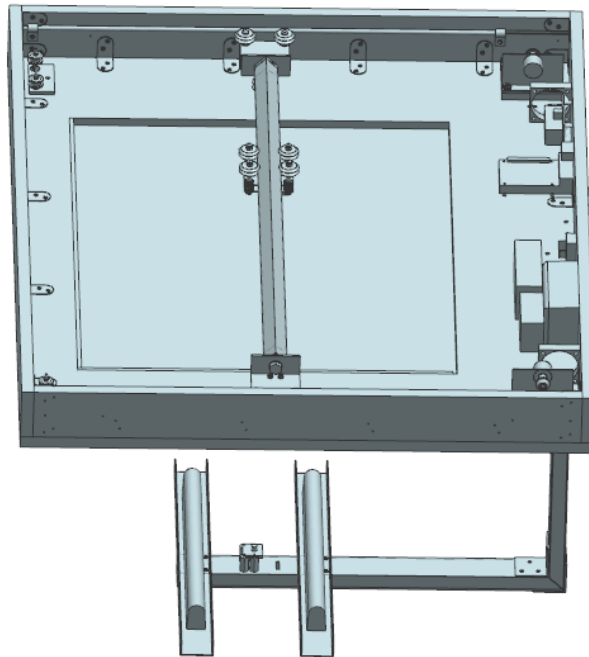


Figura 69. Captura del 3D vista des de sota. (Font elaboració pròpia)



Figura 70. El prototip vist des de sota sense la tapa inferior. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

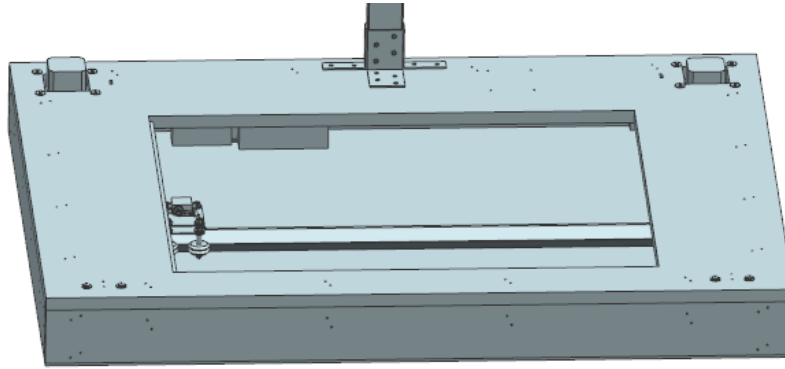


Figura 71. Captura del 3D, vista superior de la caixa. (Font: Elaboració pròpia)



Figura 72. Caixa vista des de dalt. (Font: Elaboració pròpia)

El material per a la estructura és el tauler contraxapat de 15mm de gruix, que es pot trobar en qualsevol botiga de venta de fusta. Però és recomanable comprar-ho en un lloc on també la tallin perquè les planxes de fusta son força grans i en un magatzem de fustes s'ha de comprar sencera.



Figura 73. Detall d'un vèrtex de la caixa del prototip. (Font: Elaboració pròpia)

Escaires estructurals i d'ajust:

En aquest projecte s'utilitzen escaires en comptes d'unions cargolades directament entre fustes o unions encolades per tres motius:

- Donar resistència a la estructura
- Permetre certa mobilitat
- Per estètica, evitant que es vegin els cargols des de l'exterior

Són uns escaires petits que medeixen 30x30x20 mm es poden trobar en ferreteries.



Figura 74. Escaires estructurals i d'ajust. (Font: Elaboració pròpia)

3.3.2 PROCÉS DE DISSENY. EL SUPORT DEL MOTOR.

El primer disseny que es va plantejar va ser utilitzar un suport comercial específic per el motor NEMA 17.

Aquet suport està pensat per ser collat de la manera que es pot observar en la imatge. Aquest sistema va bé quan es disposa d'espai com és el cas de les impressores 3D, però en aquest projecte en que totes les peces s'han de muntar dins d'una caixa de poca altura, aquest tipus de muntatge no era viable ja que requeria fer una caixa massa alta.



Figura 76. Suport per al motor NEMA 17 en "L". (Font: <https://www.amazon.es>)

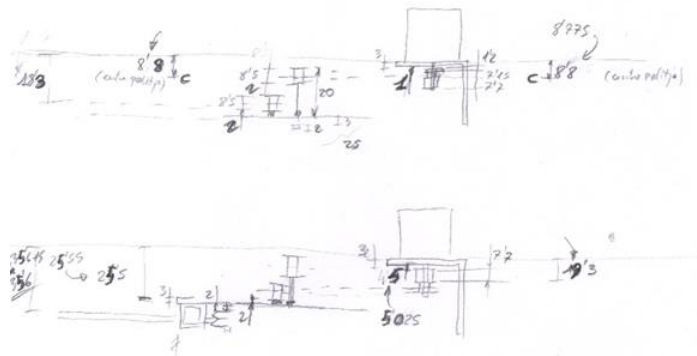


Figura 76. Esquema de muntatge dels motors sobre els suports en "L". (Font elaboració pròpia)

Posteriorment es va plantejar colocar els motors sobre un suport amb la forma que es pot veure a l'esquema de sota.

Aquest suport es collaria a la base de la caixa i permetria moure el motor endavant i endarrere per posar i treure la corretja i també tensar-la.

Aquesta peça requeria molta elaboració i obligava a que la base de la caixa fos una fusta molt gruixuda, així que la idea es va acabar modificant.

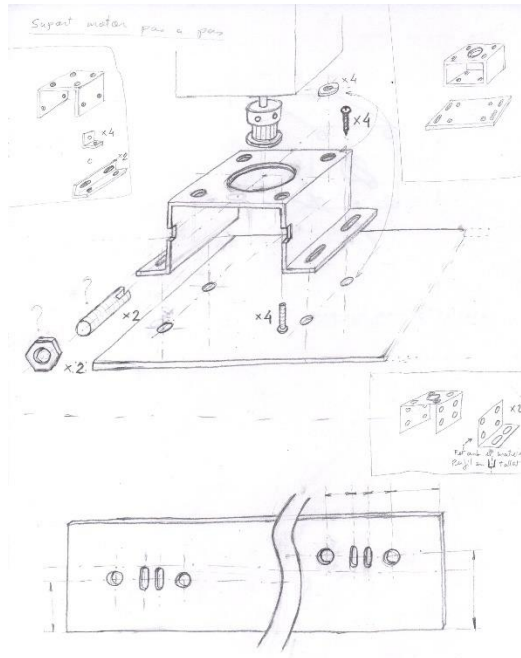


Figura 77. Esquema del disseny del suport del motor exterior i regulable, (Font: Elaboració pròpia)

Més endavant i després de decidir la estructura de la caixa, que es basava en tenir una tapa superior de fusta gruixuda que feia la funció d'estructura principal, es va decidir que la millor opció era collar els motors en na platina i aquesta platina collar-la a la estructura principal.

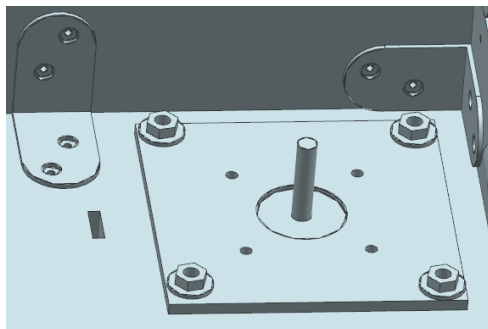


Figura 78. Detall del suport del motor collar a l'estructura principal. Vista des de sota. (Font: Elaboració pròpia)

Es van prendre mides dels components i la geometria i es va veure que la platina s'hauria de collar per sota de la estructura principal

En aquest cas, els motors haurien de travessar la estructura principal des de la part superior per acabar sortint a l'interior de la caixa on es troba el mecanisme i es va considerar que no hi havia problema en que els motors sortissin per la part superior de la caixa si així es podia reduir l'altura d'aquesta i simplificar el disseny.

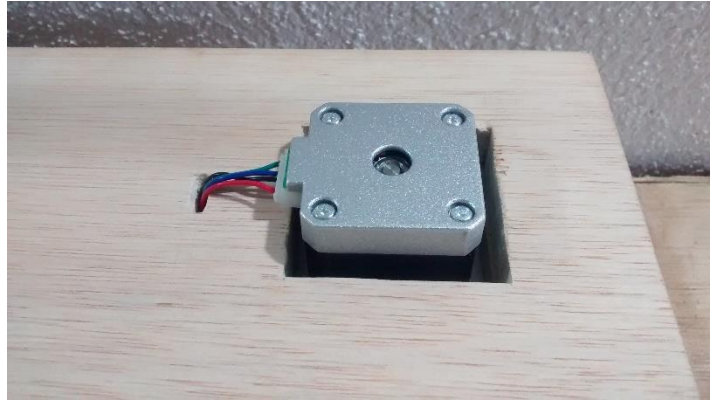


Figura 79. Detall del motor travessant el forat mecanitzat en la estructura principal. (Font: Elaboració pròpia)

3.3.3 PROCÉS DE DISSENY. LES GUIES FIXES I ELS SEUS SUPORTS.

Un dels primers plantejaments va ser utilitzar patins de perfil rectangular de tefló que llisquessin per l'interior d'un carrils fets amb perfils d'alumini o acer recoberts amb una lamina de plàstic amb una composició de tipus "Sandwich" com es pot veure en els esquemes de la imatge inferior.

Es va veure que aquest sistema podria comportar problemes de funcionament a la pràctica, ja que al no estar suportada sobre elements que siguin capaços de lliscar sota una elevada pressió com ara coixinets lineals o rodes la guia mòbil es podria travessar i quedar clavada. Així que aquesta idea es va descartar.

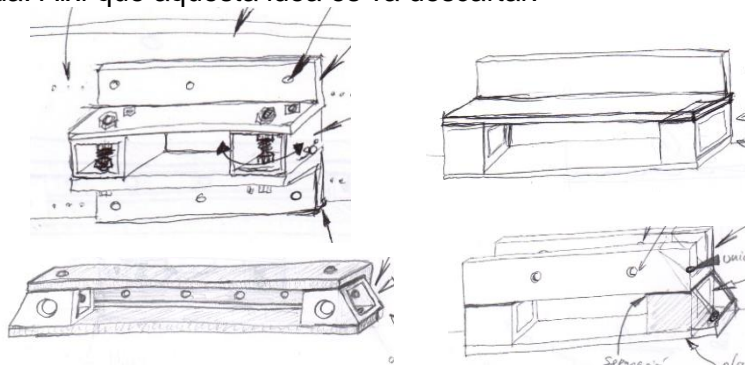


Figura 80. Diversos esquemes de models de guia fixe tipus "Sandwich". (Font: Elaboració pròpia)

Quan es va fer el plantejament d'utilitzar guies cilíndriques auto suportades, es a dir, que tinguessin prou resistència per elles mateixes com per no necessitar suport extern al llarg de tota la guia, la idea era utilitzar uns suports comercials força comuns per a guies fixes d'impressores 3D i fresadores.



Figura 73. Guies comercials cilíndriques d'acer amb els seus respectius suports i patins. (Font:

Quan es va comprovar que la mida de les guies autosuportades econòmiques que hi havia al mercat eren massa curtes es va haver de canviar el disseny i per tant també el disseny dels seus suports.

Amb aquest propòsit es van voler dissenyar unes guies que es poguessin obtenir de perfils extruïts comercials estandar, que no fossin venuts com a guies per a màquines de precisió sinó com a material de construcció o bricolatge genèric de manera que el seu cost seria molt menor.

En aquest punt ja s'havia decidit utilitzar les rodes V-Slot i es va començar a buscar un perfil en forma d'escaire a 90° que d'alguna manera es pogués collar a la paret. Com que no es van trobar aquest tipus de perfils es van començar a dissenyar sistemes per unir de perfils angulars estandar a les parets de fusta de la caixa, de manera que les rodes poguessin circular per la seva aresta central.

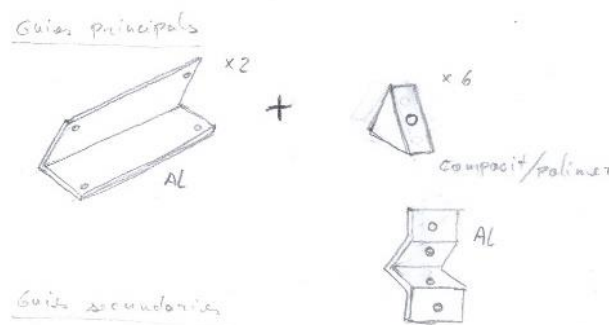


Figura 81. Esquema de guia en forma d'escaire i sistema d'unió. (Font: Elaboració pròpia)

Aquest sistema es va trobar massa laboriós i complicat de fabricar, el que es traduïa en poca precisió si es feia per mitjans casolans o un preu molt elevat si s'encarregava a un centre de mecanitzat, així que es van seguir buscant noves idees.

En el moment en que es va plantejar que tot el conjunt aniria tancat a l'interior d'una caixa, va sorgir la idea d'utilitzar les propies parets com a guies fent servir rodes acabades en punta. La idea semblava bona però de seguida es va comprovar que no ho era tant. Si es feia servir aquest mètode, la caixa s'hauria de poder ajustar en horitzontal com en vertical, una situació complicada a la pràctica, ja que si es mouen les parets laterals cap a l'interior les parets superior i inferior no es poden moure.

La solució seria utilitzar perfils en forma d'escaire i collarlos a les parets per fer de guia superior. Si es feia això llavors s'haurien de collar una altra parell d'escaires a la part inferior per igualar les mides. Això no era realment un problema, però sent sincer, no se'm va acudir en aquell moment.

El problema més important que plantejava aquest sistema era la quantitat de rodes que s'havien d'utilitzar i la posició en que s'havien de col·locar. Es necessitaven 4 rodes per costat, és a dir un total de 8 rodes només per la guia mòbil, i estan col·locades en un angle de 45°, per tant la peça que els suporti no es pot treure d'un perfil extruït comercial, sinó que s'ha de plegar una planxa a la mida desitjada.

En aquell moment es volia evitar haver de recórrer a tallers de mecanitzat i a part vaig descobrir la roda V-Slot, una roda comercial força barata amb unes dimensions adequades i una geometria que permetia utilitzar-ne només quatre en comptes de vuit si es trobava la guia adequada, així que es va descartar la idea d'utilitzar les arestes interiors de la caixa com a guia.



Figura 82. Esquema de la idea d'utilitzar la pròpia caixa com a guia. (Font: Elaboració pròpia)

Un cop descoberta la roda V-Slot i descartades les opcions comentades anteriorment, el repte era trobar una guia adequada per aquesta roda.



Figura 83. Rodes tipus V-Slot. (Font: <https://www.amazon.es>)

Es buscava una guia que no fos autosuportada i es pogués apoiar sobre les parets laterals de la caixa pels motius que s'han comentat anteriorment. També era necessari que encaixés amb les mides del canal intern de la roda i que no fós una guia que es comercialitzés com a tal per així reduir el preu.

Es van plantejar diverses idees que o bé no eren ideals o bé no es podien trobar al mercat en les dimensions necessàries o bé eren massa cares.

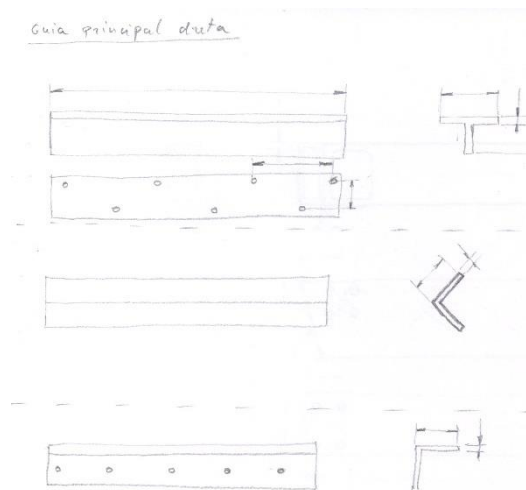


Figura 84. Diverses formes per a les guies fixes. (Font: Elaboració pròpia)

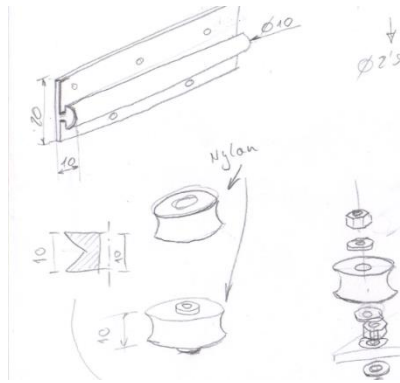


Figura 85. Esquema de guia en forma de bolet i rodes compatibles. (Font: Elaboració pròpia)

La idea que va sorgir finalment va ser utilitzar un tub de 6mm de diàmetre exterior, encaixava perfectament a l'interior del canal de la roda i a més al cer una guia tubular contra un canal en forma de V generava un contacte puntual que permetia un desplaçament suau.

Ara el repte estava en dissenyar un sistema que suportés aquest tub i l'hi aportés resistència estructural ja que per si sol no la tenia.

La solució va ser molt senzilla i elegant, tallar una canal de punta a punta de cada una de les parets laterals on s'assentaria el tub. Aquest canal havia de tenir una amplada inferior al diàmetre exterior del tub i una profunditat major que el seu radi, de manera que el tub quedés encaixat entre les dues arestes superiors del canal i no es pogués desplaçar ni cap a dalt ni cap a baix ni cap a l'exterior. Un cop muntada la estructura mòbil amb les rodes sobre aquest carril, la pròpia força entre les dues guies no permetria que el tub sortís del canal caïent cap a l'interior de la caixa, ja que per fer-ho hauria de desplaçar les rodes cap a l'interior i això no era possible ja que les rodes estarien connectades a la guia mòbil que a la vegada s'apoiava sobre la guia fixe oposada suportada per la corresponent paret lateral.

Aquesta es va veure com la solució perfecte ja que era simple de fabricar i de muntar a part de molt barata i elegant, així que es va decidir aplicar-la al prototip.

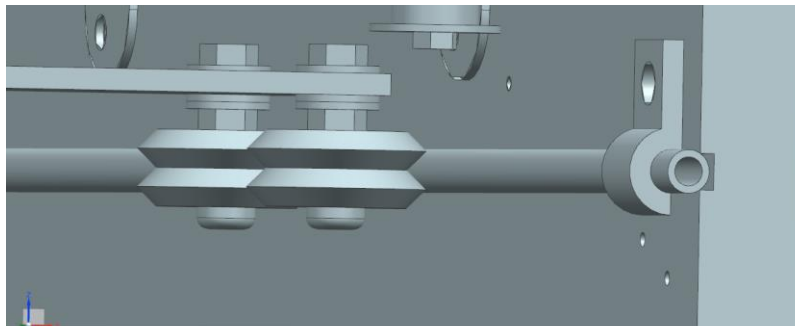


Figura 86. Captura del 3D. Vista d'una guia fixe, els suports i les rodes a sobre. (Font: Elaboració pròpia)



Figura 87. Foto del prototip. Detall de la guia fixe amb les rodes a sobre. (Font: Elaboració pròpia)

Per evitar que el tub que feia de guia fixe sortís del canal on estava col·locat, es necessitava algun element que l'aguantés. Es podia utilitzar qualsevol cosa, ja que aquest element no ha de fer força, només mantenir la guia en el seu lloc per facilitar el muntatge.

La millor manera de fer-ho era utilitzar grapes per a tub de 6mm disponible a qualsevol ferreteria.

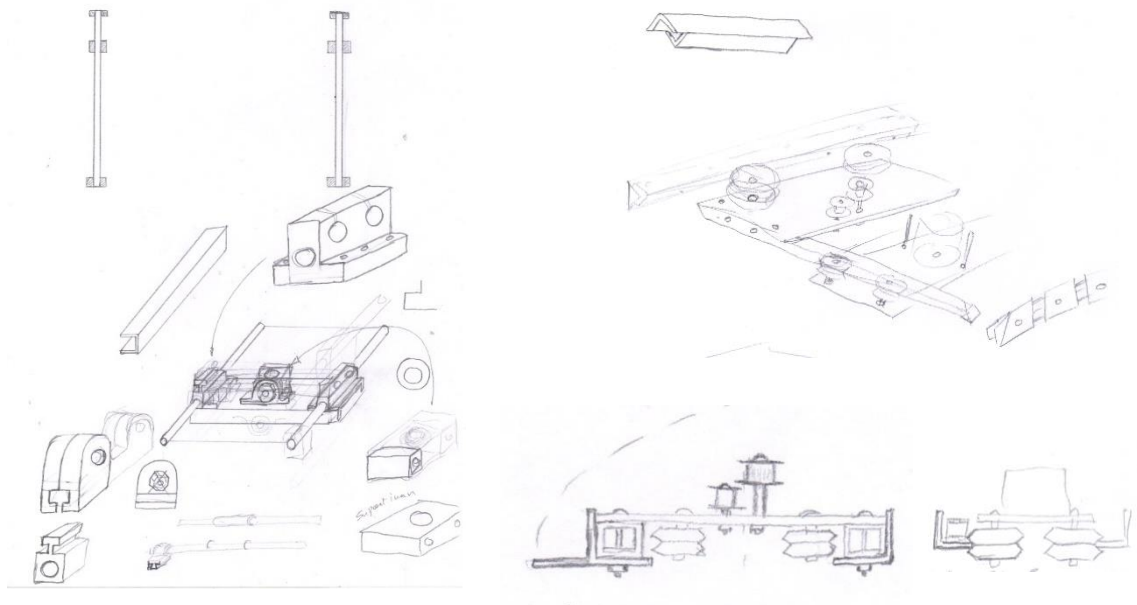
S'han utilitzar les grapes d'una pota perquè no és necessari que facin força i amb un sol cargol es facilita el muntatge.



Figura 88. Grapa per tub rodó Ø6 mm. (Font: <https://www.ferreteriasindustriales.es>)

3.3.4 PROCÉS DE DISSENY. GUIA MÒBIL.

Inicialment es van plantejar molts tipus de dissenys per a la guia mòbil, que van anar evolucionant. Alguns són els següents:



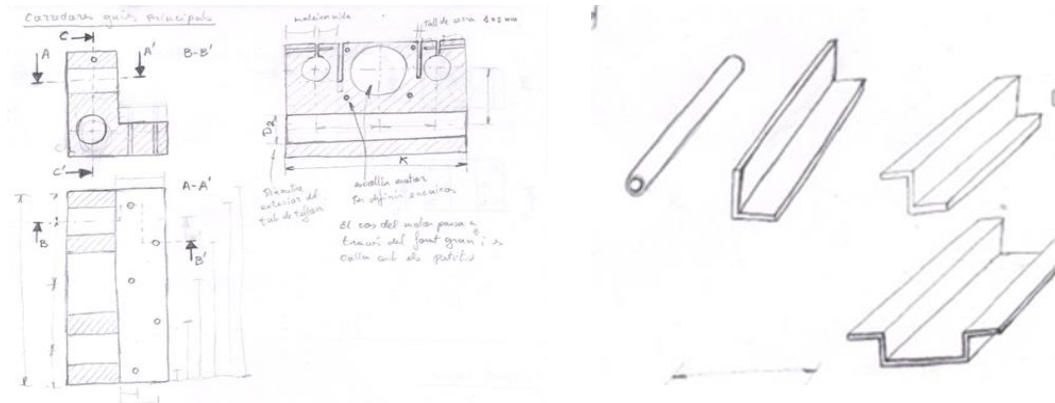


Figura 89. Diferents propostes per a la guia mòbil. (Font: Elaboració pròpia)

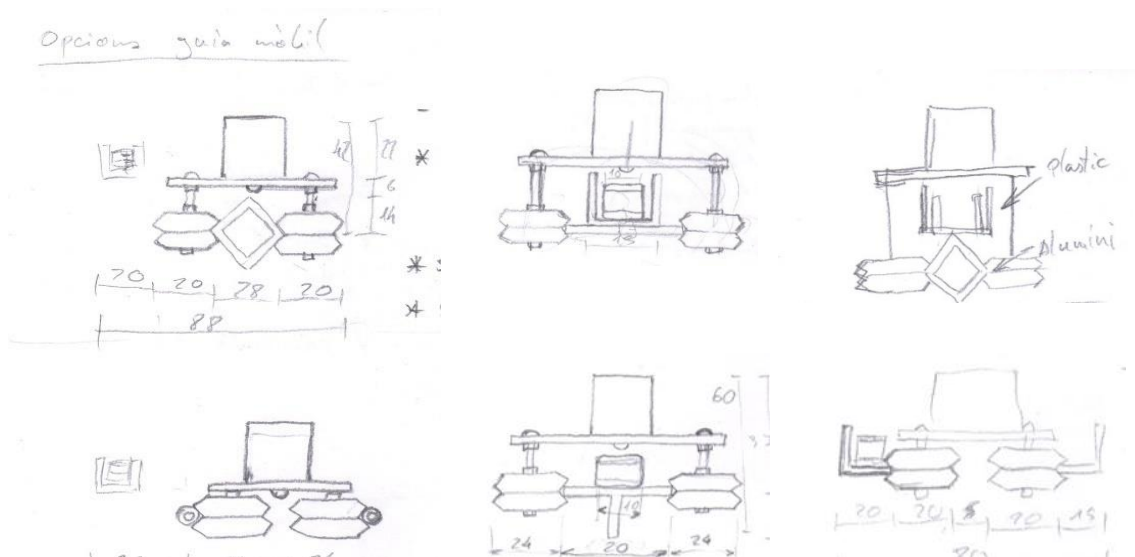


Figura 90. Diversos plantejaments respecte el muntatge a utilitzar en la guia mòbil. (Font: Elaboració pròpia)

Tub quadrat d'alumini 20x20

Per a la guia mòbil es requeria un sistema resistent, simple i lleuger, ja que era una estructura mòbil, per tant havia de tenir el mínim pes possible per estalviar esforç als motors, a la vegada havia de servir de guia per a les rodes V-Slot comentades anteriorment i havia de ser robust per resistir la torsió generada per les corretges sobre les politges muntades en els dos extrems de la guia.

Finalment es va decidir utilitzar un tub quadrat d'alumini per a la estructura-guia.

Aquesta solució agrupava tots els requisits:

- Lleugeresa: al ser un tub quadrat d'alumini de poc gruix.
- Resistència: la secció quadrada l'hi confereix una elevada resistència a la torsió en totes direccions.
- Guia: els seus costats a 90° són una bona guia per a la roda V-Slot.
- Econòmic: comparat amb les guies fetes expressament per a impressores 3D o

màquines CNC industrials.



Figura 91. Tub quadrat d'alumini 20x20. (Font: <http://www.leroymerlin.es>)

Inicialment es preveia utilitzar una secció més petita per reduir encara més el pes i el volum, però finalment es va escollir la secció 20x20 per adaptar-se als taps per tub quadrat roscats disponibles a les ferreteries, com s'explica a continuació.

Tap roscat de plàstic per a tub quadrat de 20x20

Als extrems del tub quadrat s'hi ha d'unir unes peces en forma de "L" que faran de suport a diverses rodes i politges. La manera més senzilla de unir-ho és mitjançant un tap de plàstic quadrat fet especialment per aquest tipus de tubs, i que conté en el centre una rosca que el travessa de punta a punta. Aquests taps s'utilitzen normalment com a base de les potes quadrades de a les quals es pot roscar un sistema d'ajust de l'altura roscat i evitar així que el moble balli.



Figura 92. Inserts per tub quadrat de 20x20mm amb forat roscat M8. (Font: Elaboració pròpia)

En aquest cas es va utilitzar per unir les dues peces mitjançant un cargol M8 que travessava la peça en forma de "L" i es collava al tap roscat prèviament col·locat a pressió a l'interior del tub quadrat.

El perquè d'utilitzar un tub i un tap quadrat de 20x20 és ni més ni menys que un tema de catàleg. A cap lloc venien taps quadrats roscats de menys de 20x20, així que vaig adaptar la mida del tub a la mida del tap.

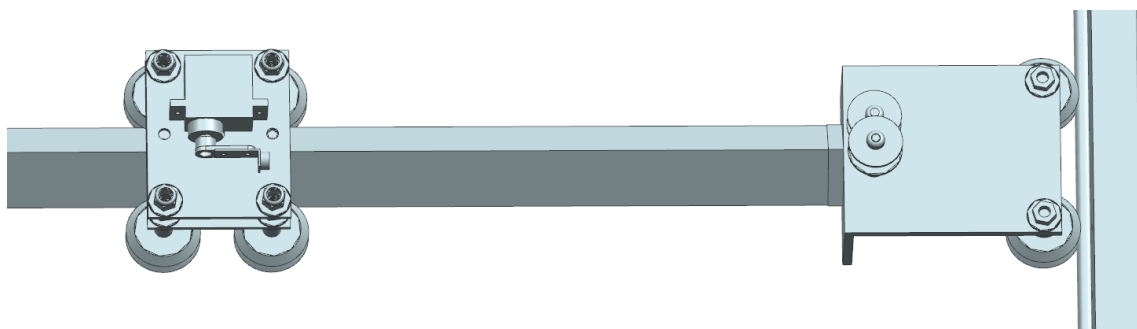


Figura 93. Costat dret dels sistema de guia mòbil en el 3D. Vista superior. (Font: Elaboració pròpia)

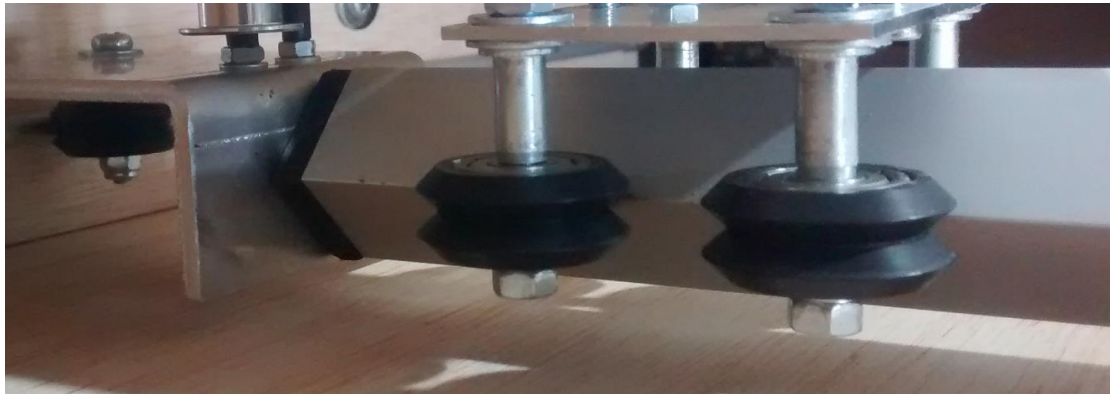


Figura 94. Foto del costat esquerre de la guia mòbil en el prototip. Vista lateral. (Font: Elaboració pròpia)

3.3.5 PROCÉS DE DISSENY. SUPORT DE LES POLITGES EN LA GUIA MÒBIL.

El sistema de muntatge de les politges mòbils va portar força feina i es van fer molts dissenys excessivament complicats de fabricar avans d'arribar al disseny final.

Inicialment es vadissenyar la posició relativa entre les dues politges de manera que es pogués eliminar el moment torsor en l'eix vertical sobre la guia mobil i evitar així que aquesta pogués sortir de les guies i descarrilar.

Per aconseguir aquest objectiu es va buscar alinear les dues corretges en el mateix pla vertical i el disseny va quedar de la manera que es pot veure en la imatge inferior.

En la imatge es pot veure una politja més gran que l'altre perquè van ser les primeres que es van comprar, la petita correspona una politja plana i la gran a una dentada, però més endavant es va veure que es venien politges planes de la mateixa mida que les dentades de 20 dents i el disseny va canviar la politja petita per una de la mateixa mida que la gran però el concepte seguia sent el mateix.

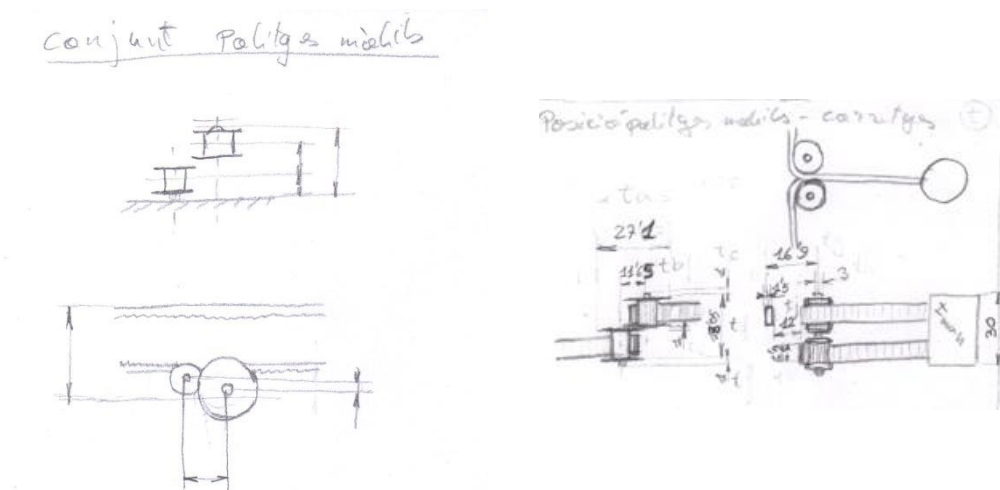


Figura 95. Esquema de muntatge de les politges mòbils. (Font: Elaboració pròpia)

A partir d'aquesta distribució i aquesta geometria bàsica es va començar a dissenyar els sistemes de muntatge físics que les podrien allotjar, a la vegada que aquest sistema s'unia a la guia mòbil i a les rodes que s'havien de desplaçar per la guia fixe.

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

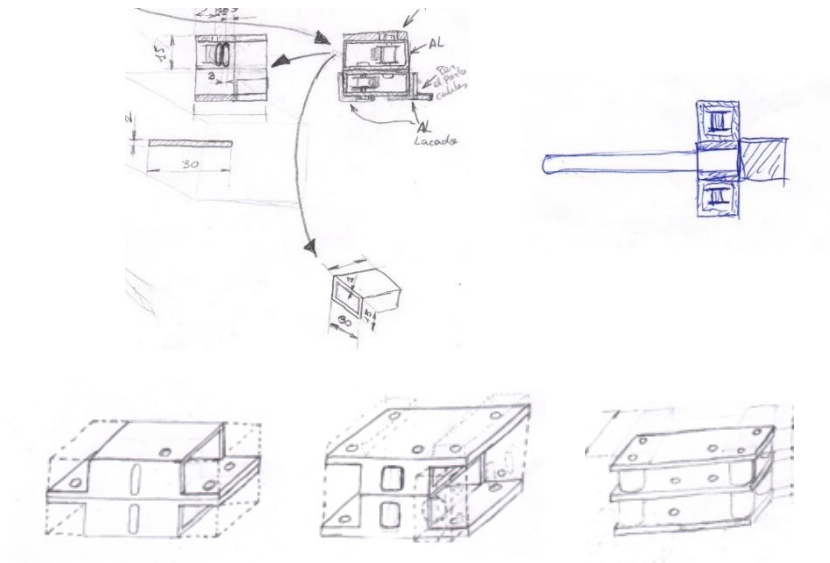


Figura 96. Diversos dissenys per allotjar les politges i unir-les a la guia mòbil i a les rodes de la guia fixa a la vegada que havien de permetre el pas de les corretges. (Font: Elaboració pròpia)

Aquests dissenys es van descartar i es va començar a desenvolupar un disseny basat en un bloc de fusta de dues parts amb allotjaments mecanitzats per les politges.

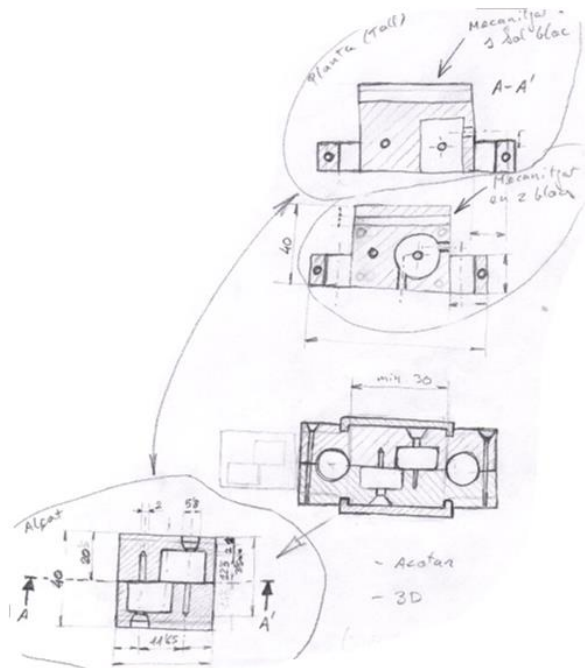


Figura 97. Disseny del suport de les politges mòbils en base a un bloc de fusta mecanitzat en dues parts. (Font: Elaboració pròpia)

Aquest disseny també es va descartar per ser massa complex en favor d'un disseny molt simple i eficient en que les politges es collaven a una xapa amb el propi cargol que les aguantava i per un sol costat. Aquest disseny fa que el cargol hagi d'aguantar torsió a diferencia dels dissenys anteriors però els esforços que se l'hi apliquen no sobrepassen la seva resistència.

Seguint la mateixa idea a aquesta xapa se l'hi podien unir les rodes amb el mateix sistema que al unir les politges.

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

El punt més complex era com unir-la a la guia mòbil. Al final, quan es va decidir utilitzar una guia mòbil quadrada, es va dissenyar una peça que s'insertava en la guia mòbil i es collava en aquest suport de politges mitjançant quatre cargols. Però es va veure que ja existia un insert similar, encara que només tenia un cargol central. Amb un sol cargol central hi havia la possibilitat de que la guia mòbil rotés en l'eix X si se l'hi aplicaven grans esforços, però davant l'estalvi econòmic es va voler provar i si fos necessari encolar les unions per evitar la rotació.

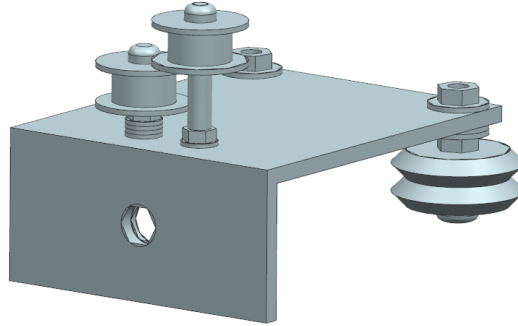


Figura 98. Suport de les politges mòbils amb les politges i les rodes muntades. Captura del 3D.
(Font: Elaboració pròpia)

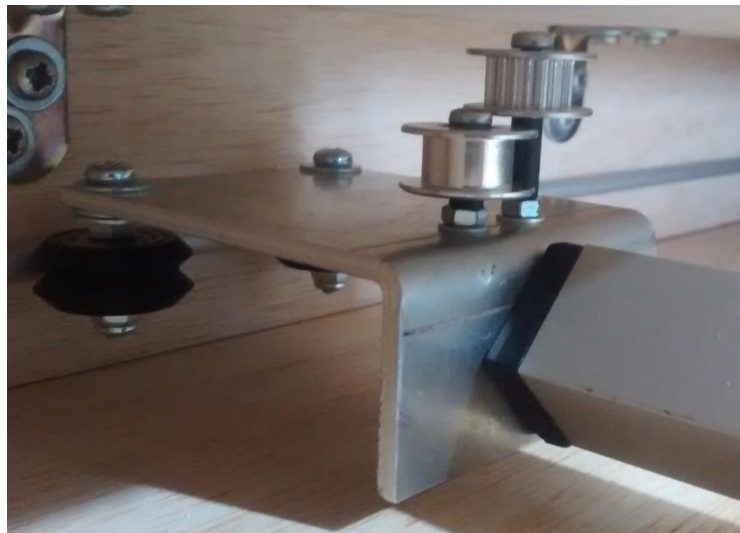


Figura 99. Foto del suport de les politges mòbils en el prototip. (Font: Elaboració pròpia)

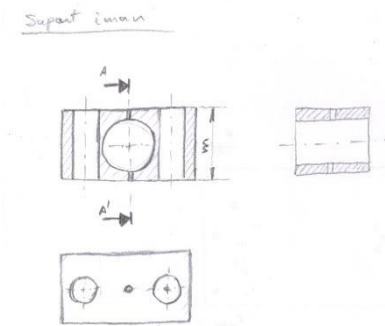
3.3.6 SISTEMA MÒBIL 2D

El sistema mòbil 2D és la part del mecanisme que es pot moure en dues dimensions, és el que es podrà posicionar sota qualsevol punt del taulell per permetre al servomotor capturar la peça amb l'imant.

Les etapes de disseny van ser les següents:

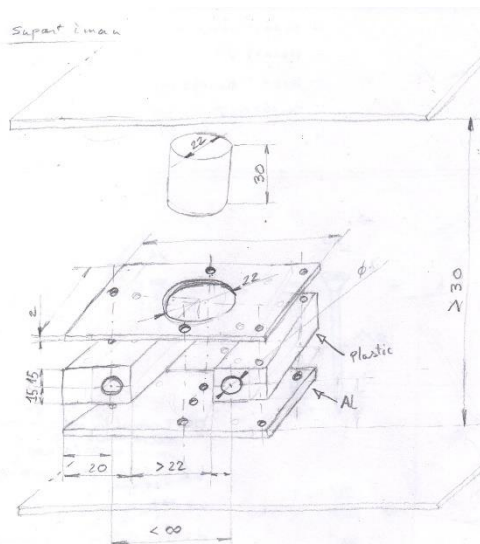
Inicialment un bloc de fusta on s'hi mecanitzaria un forat gran central del diàmetre de l'electroimant i un forat perpendicular més petit a cada costat d'aquest per on passarien unes guies cilíndriques.

En els forats per les guies s'hi hauria d'introduir un tub de tefló entre fusta i guia per permetre el desplaçament sense fricció.



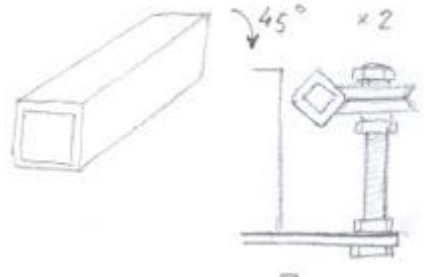
Després es va evolucionar el disseny a dues planxes superposades amb dos perfils rectangulars de tefló foradats entremig que permetien el pas de les guies. Aquest disseny permetia a l'imant refrigerar-se ja que durant les proves es va comprovar que generava molta calor.

A part d'això, permetia ajustar l'altura de l'electroimant mitjançant un cargol collat a la platina inferior que s'arribaria a collar al forat roscat de la part inferior del cargol.



Posteriorment quan es va decidir no utilitzar guies cilíndriques, el disseny va evolucionar a una sola platina d'on sortien uns cargols amb unes rodes que s'encarrilaven en dues guies que transcorrien lateralment per els dos costats de l'electroimant.

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada



En la següent etapa del disseny s'havia decidit utilitzar una sola guia central quadrada i el disseny va mutar a una platina de la qual sortien quatre cargols cap avall amb una roda a cada cargol que s'unien a la guia.

Posteriorment a aquesta platina se l'hi va afegir una segona per augmentar la seva resistència mecànica i es va obtenir així el disseny final que es pot veure en la imatge:

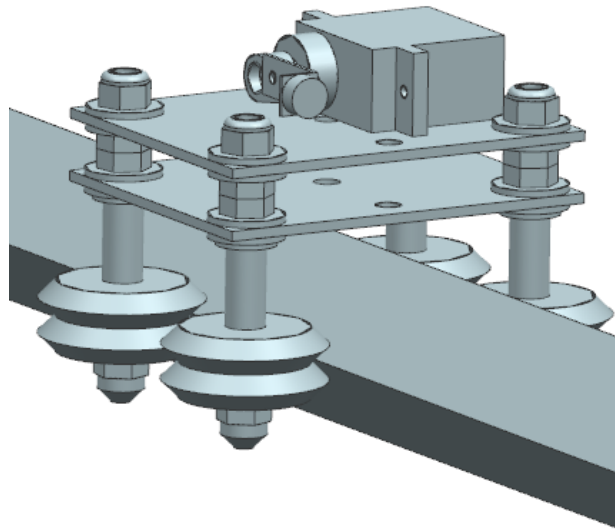


Figura 100. Vista de l'actuador 2D. Captura del disseny 3D. (Font: Elaboració pròpia)

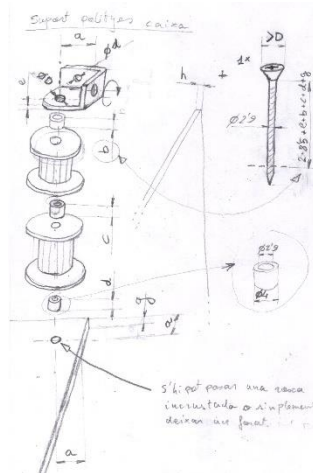


Figura 101. Vista de l'actuador 2D. Foto del prototip. (Font: Elaboració pròpia)

3.3.7 PROCÉS DE DISSENY. EL SUPORT DE LES POLITGES FIXES.

Un disseny inicial era collar-les a la fusta utilitzant cargols de fusta amb l'ajuda d'escaires de 3 costats. Aquest disseny plantejava dos problemes molt importants:

- El primer i més important el forat de les politges esta fet per a cargols mètrics, i els cargols de fusta no hi encaixen.
- El segon, l'escaire de 3 costats serveix per collar en una cantonada, però les politges no es col·locaran a una distancia que correspongui a la d'un escaire comercial, per tant s'hauria de fabricar expressament foradant doblegant i soldant xapa, cosa que no val en absolut la pena.



Posteriorment, quan la resta del disseny ja estava molt més avançat i ja es preveia l'ús d'una estructura plana elevada com a estructura principal, es va preveure de collar dos cargols mètrics directament a la estructura.

Aquest disseny era viable, però per mantenir la estètica es va preferir collar les politges a una pletina utilitzant els cargols mètrics, i aquesta pletina collar-la a la estructura principal des de sota utilitzant cargols de fusta, de manera que des de l'exterior no fos visible cap cargol com es pot veure en la següent imatge:

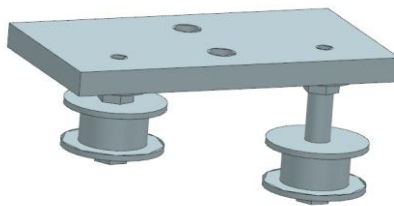


Figura 102. Placa de suport de les politges fixes amb les politges muntades. (Font: Elaboració pròpia)



Figura 103. Politges fixes muntades en el prototip (Font: Elaboració pròpia)

3.3.8 SISTEMA MÒBIL 3D / SISTEMA DE SUBJECCIÓ MAGNÈTIC

Inicialment no estava previst un sistema mòbil en la tercera dimensió, es a dir, que es pogués verticalment. Els dissenys inicials preveien utilitzar només un sistema mòbil en 2D que pogués l'actuador a una altura constant per sota el taulell.

La subjecció magnètica de les peces en aquest cas s'havia de realitzar mitjançant un electroimant.



Figura 104. Electroimants de 2'5, 5, 10, 20 i 500 N de força consecutivament amb els que es van fer proves de subjecció electromagnètica (Font: <https://www.amazon.es>)

Es van realitzar varies proves però finalment es va comprovar que amb l'espai de que es disposava no es podia col·locar un electroimant prou potent com per poder atraure les peces a través del gruix del taulell. El problema és que aquests electroimants poden exercir aquesta força que indiquen quan estan en contacte directe amb una planxa de ferro que tanca correctament el circuit magnètic, però a través d'una fusta de 5mm de gruix la força és pràcticament inexistente.

Així que es va arribar a la conclusió que s'havia d'utilitzar un sistema mecànic que pogués acostar i allunyar un imant permanent a voluntat.

El primer plantejament i el que s'aplicarà al prototip és el d'un servomotor col·locat a la part superior de l'actuador. Aquest servomotor portarà un braç a la punta del qual es col·locarà un imant permanent de neodimi.

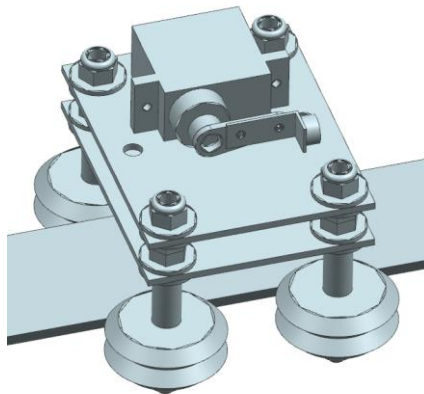


Figura 105. Muntatge del servo sobre l'actuador 2D amb l'imant de neodimi a la punta del braç.
(Font: Elaboració pròpia)



Figura 106. Motor SG90 previst d'utilitzar en el sistema de subjecció magnètica de les peces. (Font: <http://tienda.bricogeek.com>)

Quan els motors pas a pas col·loquin l'actuador sota la peça que es vol moure, es donarà l'ordre al servomotor d'eleva el braç subjectant així la peça.

Un cop la peça estigui dins l'escac d'arribada es donarà l'ordre al servomotor de retirar el braç e manera que la peça quedarà lliure i l'actuador es podrà moure a la nova posició sense arrossegar cap peça pel camí.

Els imants de neodimi utilitzats són de 6mm de diàmetre i 6mm d'altura, amb una força de subjecció aproximada de 13N.



Figura 107. Imants de neodimi de 6x6mm. (Font: <https://www.superimanes.com>)

3.3.9 PROCÉS DE DISSENY. ESTRUCTURA PER A LA CÀMERA I LA IL·LUMINACIÓ.

El disseny del sistema de il·luminació i la estructura que la suporta juntament amb la càmera parteix de les primeres proves que es van fer intentant aplicar l'anàlisi d'imatge a la foto que s'obtenia amb la càmera.

Com que l'anàlisi d'imatge depenia molt de que la càmera es mantingués en el mateix lloc on havia estat calibrada, se l'hi va muntar un suport provisional amb un filferro i una pinça de multímetre, muntat de qualsevol manera amb cinta adhesiva sobre la prestatgeria de l'habitació.

La càmera es va muntar en una posició central del tauler aproximadament 45 cm per sobre d'aquest de manera que s'eliminava el màxim possible la perspectiva.

Fent proves amb l'anàlisi d'imatge també es va veure que era absolutament imprescindible un sistema d'il·luminació directe i vertical, que sortís del punt més pròxim possible a la càmera en direcció al taulell. Sense aquest sistema d'il·luminació directa les fotos sortien massa fosques i al analitzar-les no es detectaven ni la meitat de peces. Si s'il·luminava externament per un lateral, es generaven massa ombres sobre el taulell i tampoc es podien detectar les bases de les peces.

Per obtenir aquesta il·luminació directa es va muntar provisionalment una làmpada d'escriptori a sobre del taulell el més a prop possible de la càmera.



Figura 108. Muntatge provisional inicial per a l'anàlisi d'imatge. (Font: Elaboració pròpia)

Al cap de varies proves es va veure que aquell sistema era insostenible ja que la càmera es movia molt fàcilment perquè la pinça no la subjectava bé la càmera, així que es va muntar un sistema més estable de fusta i acer amb uns forats on collar la càmera de

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

manera sòlida i tot el sistema es subjectava amb un serjant a l'estanteria, així la càmera es mantenia fixe i un cop s'havia calibrat no calia tornar-ho a fer.



Figura 109. Muntatge provisional versió 2 per a l'anàlisi d'imatge. Detall del muntatge de la càmera. (Font: Elaboració pròpia)



Figura 110. Muntatge provisional versió 2 per a l'anàlisi d'imatge. Vista del conjunt. (Font: Elaboració pròpia)

Aquest muntatge provisional es va mantenir mentre es desenvolupava tot el programa informàtic, un error important, ja que posteriorment, quan es va dissenyar el sistema mecànic, es va veure que no es podien utilitzar les mateixes peces, i un cop fabricat el prototip real, el programa s'ha de modificar parcialment i tornar-lo a calibrar. Per tant s'hagués hagut d'avançar el disseny mecànic en paral·lel al disseny electrònic i al programa informàtic per evitar aquest tipus de pèrdues de temps.



Figura 111. Muntatge provisional versió 2 per a l'anàlisi d'imatge. Detall de la il·luminació del taulell. (Font: Elaboració pròpia)

Partint de la estructura provisional muntada a l'inici del projecte, es va desenvolupar el disseny de la estructura que es fabricaria finalment i formaria part del prototip.

En aquesta s'hi instal·laria la càmera enfocant el taulell en una posició central i als dos costats s'hi instal·larien dues làmpades llargues per il·luminar el taulell uniformement.



Figura 112. Muntatge de la càmera i el sistema d'il·luminació final en el prototip. Vista des de sota. (Font: Elaboració pròpia)

El cos de la estructura estaria formada per dos tubs de perfil rectangular d'alumini amb unes dimensions de 20x40x1'5, una mida comercial molt fàcil de trobar a les ferreteries

Dispositiu automàtic per a jugar a escacs Marçal Civill Cortada

i tallers i que tenia una mida suficient tant per muntar-hi la càmera com per permetre el pas dels cables elèctrics de les làmpades i el cable de cinta de la càmera pel seu interior, a la vegada que oferia una estructura robusta.

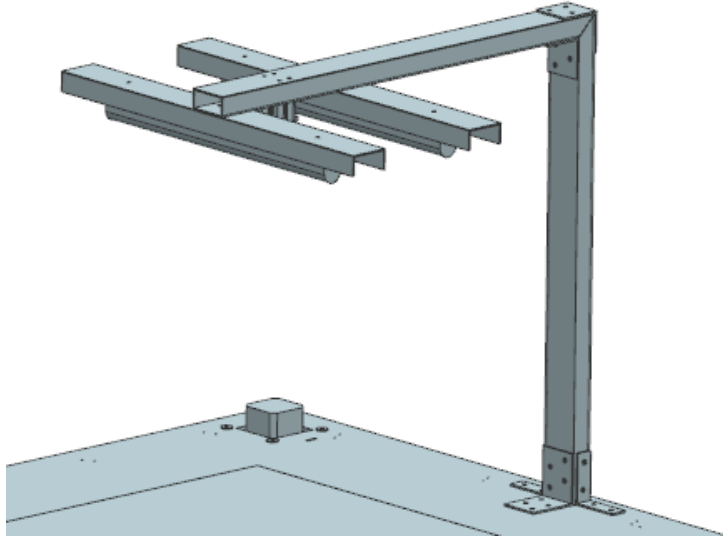


Figura 113. Vista general del diseny 3D de la estructura. (Font: Elaboració pròpia)



Figura 114. Vista general del prototip. (Font: Elaboració pròpia)

La càmera es muntaria amb cargols M2 a través d'una de les cares de la estructura fent quatre forats frans per l'altre costat per passar el tornavís.

També s'hi hauria de fer un forat rectangular per passar el cable de cinta.



Figura 115. Detall del muntatge de la càmera en el prototip. (Font: Elaboració pròpia)

Aquests dos tubs es muntarien entre ells utilitzant escaires d'acer i reblons que es reblonarien amb una reblonadora manual.

L'estructura s'uniria a la vegada a la estructura principal mitjançant escaires d'acer collats amb cargols mètrics a uns inserts fusta-M4 que prèviament s'haurien collat a la estructura principal. Això permetria desmuntar la estructura si fós necessari per al transport.

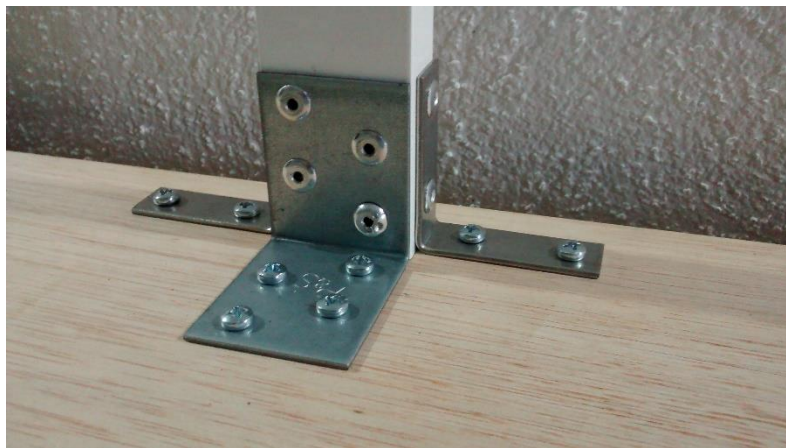


Figura 116. Detall de la unió entre la estructura de la càmera i la estructura principal. (Font: Elaboració pròpia)

Tots els cables recorren la estructura i surten per sota travessant la estructura principal, podent així connectar-se al sistema elèctric i electrònic muntats a l'interior de la caixa.

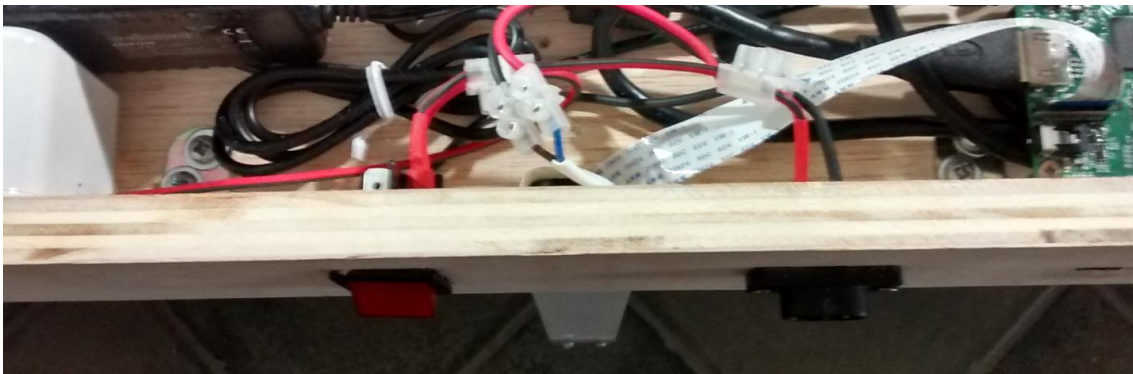


Figura 117. Detall de la sortida dels cables provinents de les làmpades i la càmera a través de la estructura principal. Vista des de sota. (Font: Elaboració pròpia)

3.3.10 PROCÉS DE DISSENY. DISSENY DEL SUPORT DELS POTENCIÒMETRES.

Aquest suport esta fabricat a partir de tallar i foradar un perfil en escaire d'alumini de 40x60x1,5 unes mides força utilitzades en tallers, però difícil de trobar en ferreteries.

Inicialment es va fer el plantejament d'unit l'eix del motor i l'eix del potenciòmetre en sèrie, però això faria que s'hagués d'augmentar l'altura de la caixa, cosa que es volia evitar per mantenir la ergonomia i la estètica del taulell. Així que es va deidir col·locar-los amb els dos eixos en perpendicular unint-los mitjançant rodes dentades a 45°.

A partir d'aquest plantejament es va dissenyar el suport del potenciòmetre de manera que les dues rodes dentades poguessin engranar correctament.

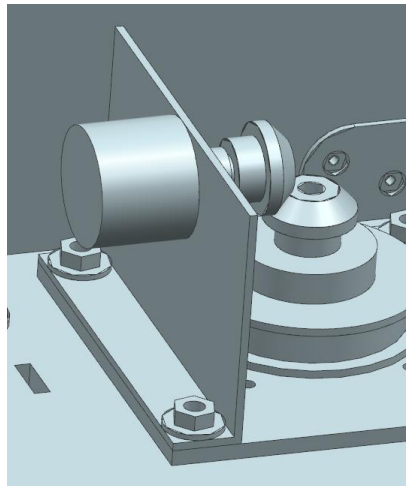


Figura 118. Vista del suport del potenciòmetre en el 3D. (Font: Elaboració pròpia)



Figura 119. Vista del suport del potenciòmetre en el prototip. (Font: Elaboració pròpia)

3.4 SISTEMES ELÈCTRIC I ELECTRÒNIC

3.4.1 ESQUEMA ELECTRÒNIC GENERAL. DIAGRAMA DE FLUX.

En aquest diagrama es representa principalment el flux de la informació i la energia que alimenta els dispositius que la generen i la controlen. En apartats posteriors es mostren les connexions específiques en cada component.

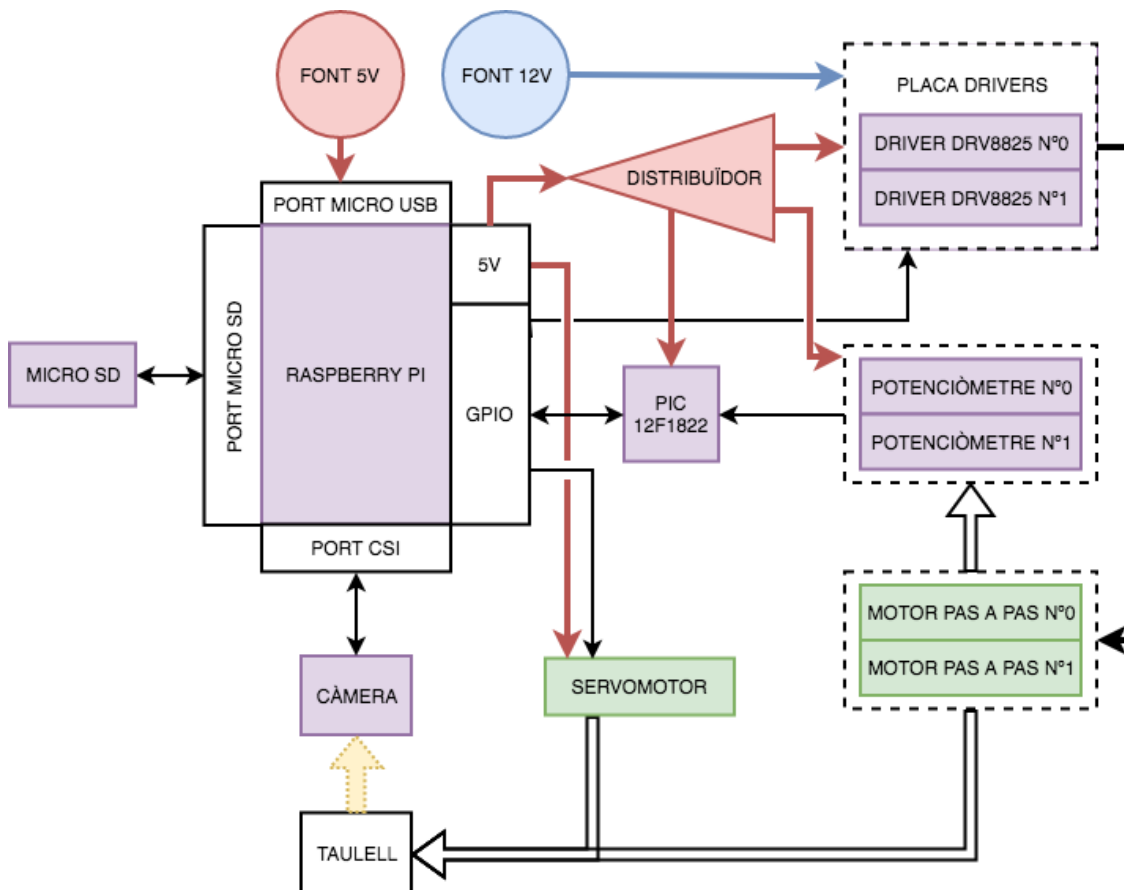


Figura 120. Esquema electrònic general. Diagrama de flux. (Font: Elaboració pròpia)

LLEGENDA:

Colors de les fletxes:

- Negres primes: Flux d'informació
- Negra gruixuda: Combinació entre informació i potència
- Vermelles: Alimentació a 5V de baixa potència
- Blaves: Alimentació a 12 volts d'alta potència
- Blanques: Actuació mecànica
- Groga: Llum des del taulell a la càmera

3.4.2 ESQUEMA ELÈCTRIC GENERAL

Per facilitar la visualització s'han marcat de color negre els cables connectats a 220V CA, en vermell els cables connectats a 5V CC i en blau els connectats a 12V CC.

Nota: Es tracta d'un esquema elèctric general, les connexions específiques es poden veure en els esquemes del punt anterior "3.3. Sistema electrònic".

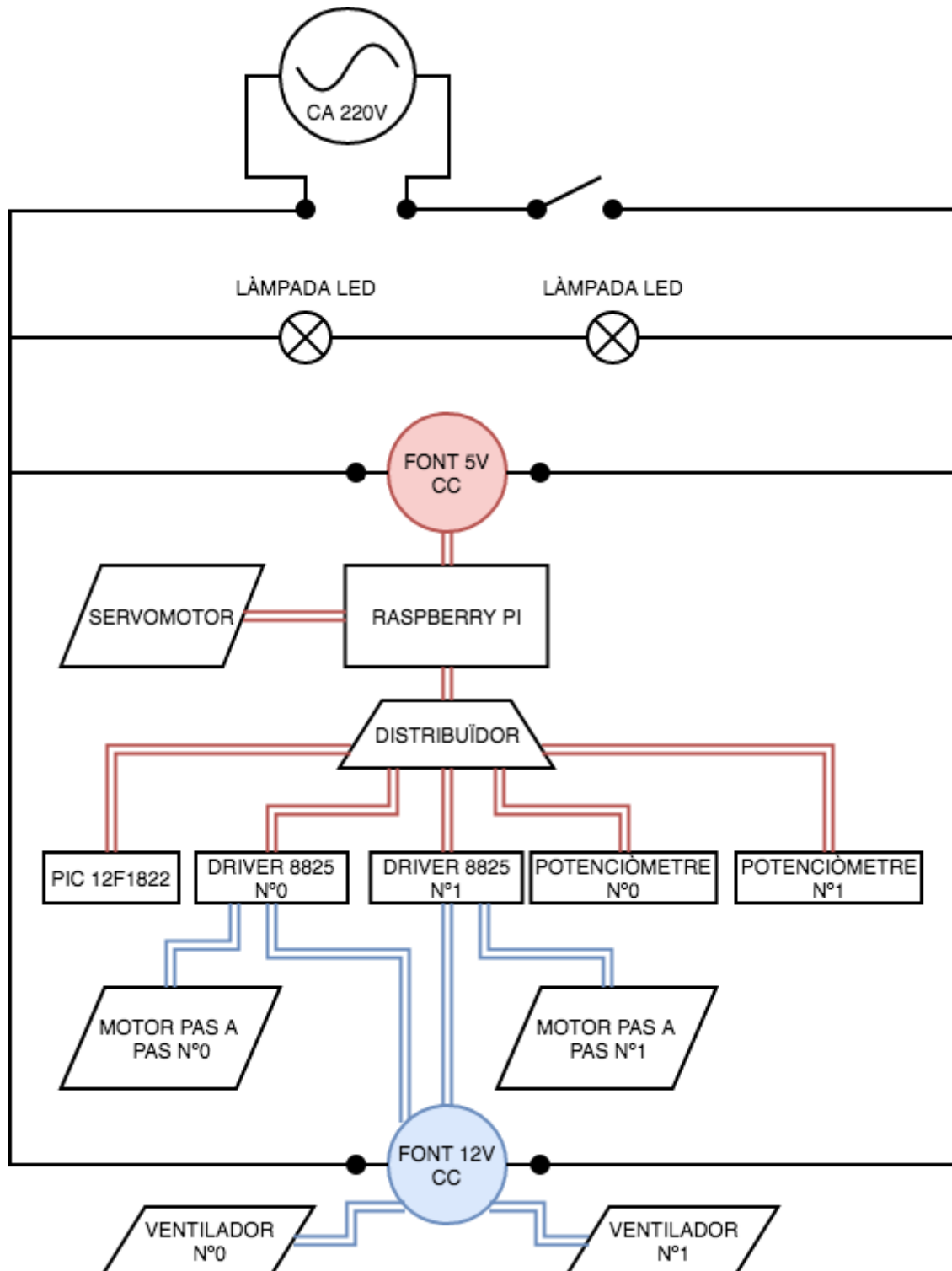


Figura 121. Esquema elèctric. (Font: Elaboració pròpia)

3.4.3 ESQUEMA DE CONNEXIONS AL GPIO DE LA RASPBERRY PI

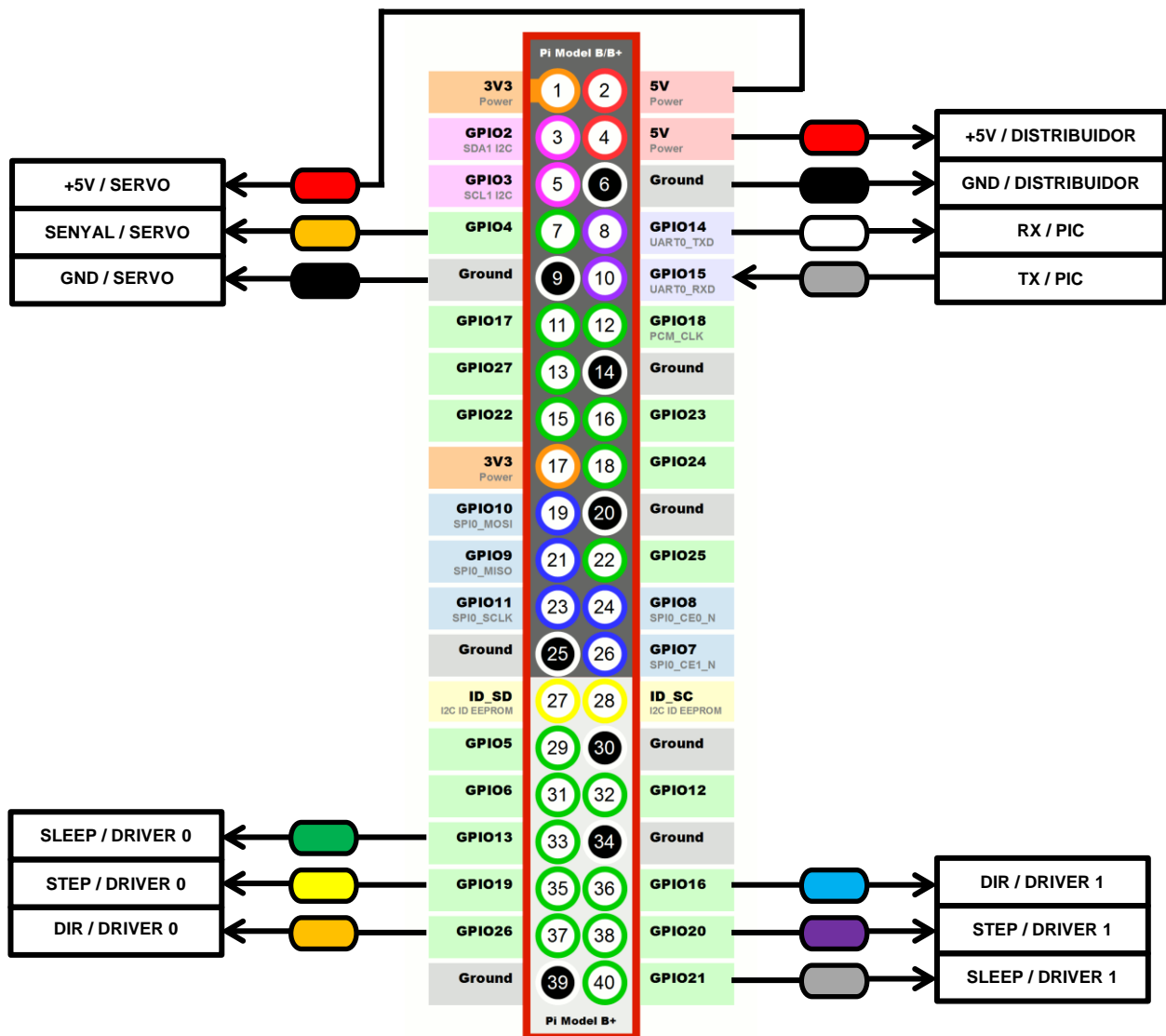


Figura 122. Esquema de connexions al GPIO de la Raspberry Pi. (Font: Elaboració pròpia)

3.4.4 CONNEXIÓ DE LA CÀMERA A LA RASPBERRY PI

La càmera només té un cable de cinta que s'ha de connectar al connector CSI que es troba al costat del connector Ethernet. La cinta té una cara blanca i una cara escrita, la cara escrita s'ha de col·locar en direcció al connector Ethernet.



Figura 123. Connexió de la càmera a la Raspberry Pi. (Font: <https://www.zapals.com>)

3.4.5 ESQUEMA DE CONNEXIONS AL DRIVER DRV8825 N°0

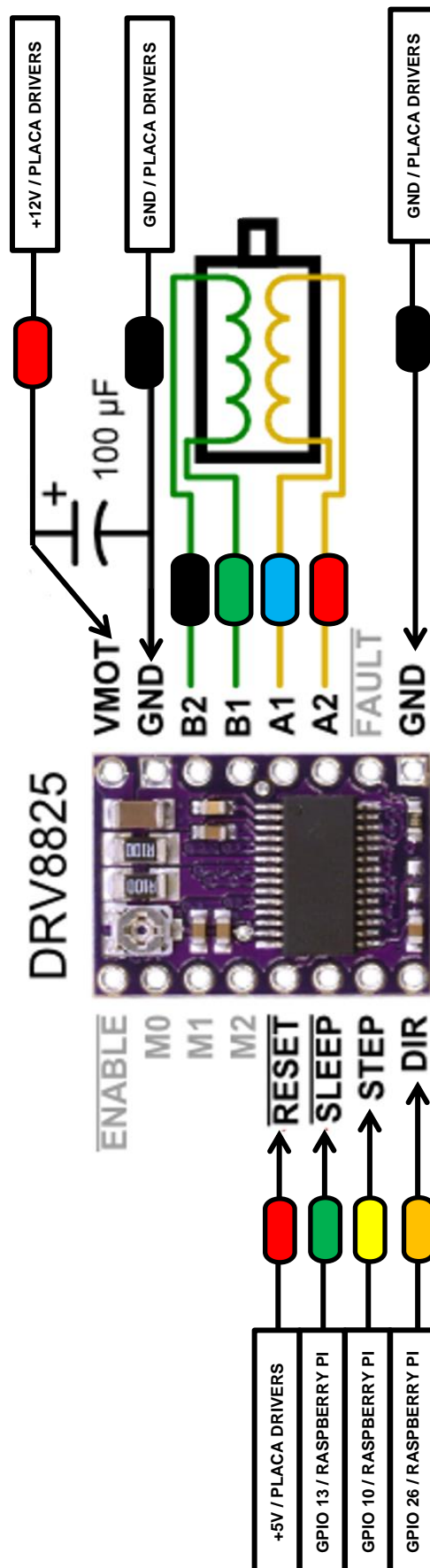


Figura 124. Esquema de connexions al driver nº 0. (Font: Elaboració pròpia)

3.4.6 ESQUEMA DE CONNEXIONS AL DRIVER DRV8825 N°1

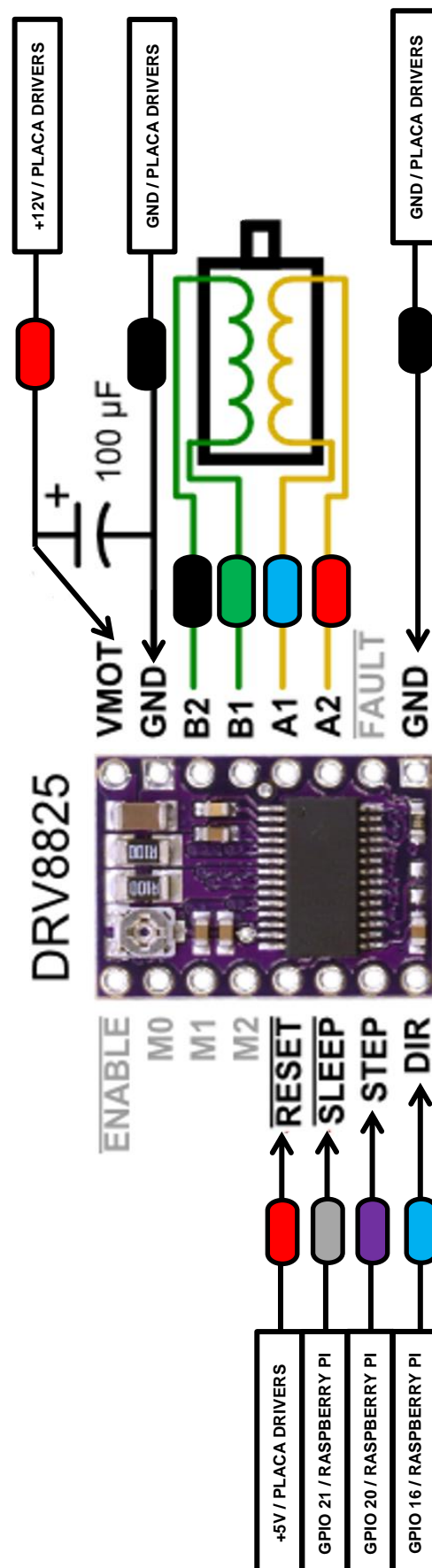


Figura 125. Esquema de connexions al driver n° 1. (Font: Elaboració pròpia)

3.4.7 ESQUEMA DE CONNEXIONS AL POTENCIÒMETRE N°0

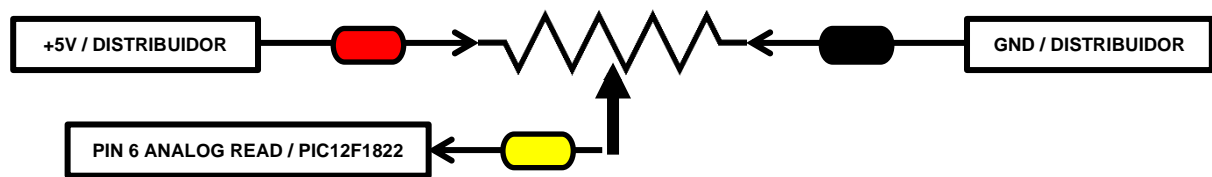


Figura 126. Esquema de connexions al potenciòmetre n°0. (Font: Elaboració pròpia)

3.4.8 ESQUEMA DE CONNEXIONS AL POTENCIÒMETRE N°1

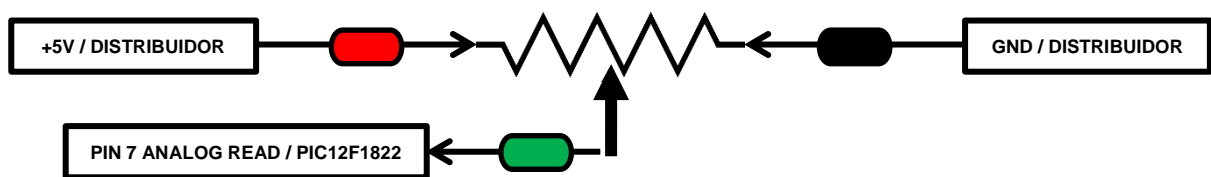


Figura 127. Esquema de connexions al potenciòmetre n°1. (Font: Elaboració pròpia)

3.4.9 ESQUEMA DE CONNEXIONS AL PIC 12F1822

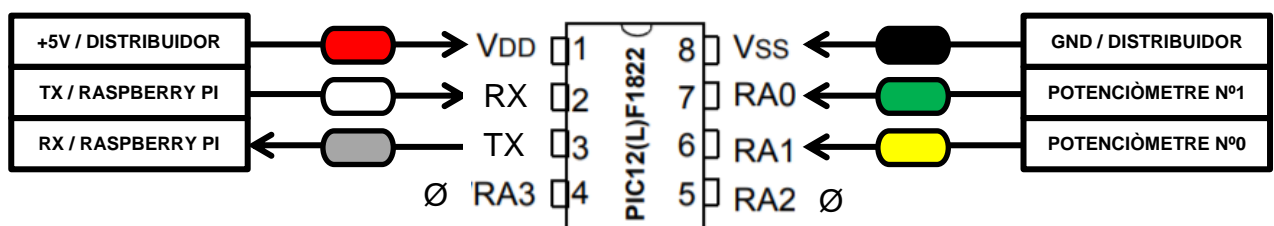


Figura 128. Esquema de connexions al PIC12F1822. (Font: Elaboració pròpia)

3.4.10 PLACA DELS DRIVERS

La placa de drivers és una placa on es poden instal·lar fàcilment els drivers i realitzar les diverses connexions necessàries.

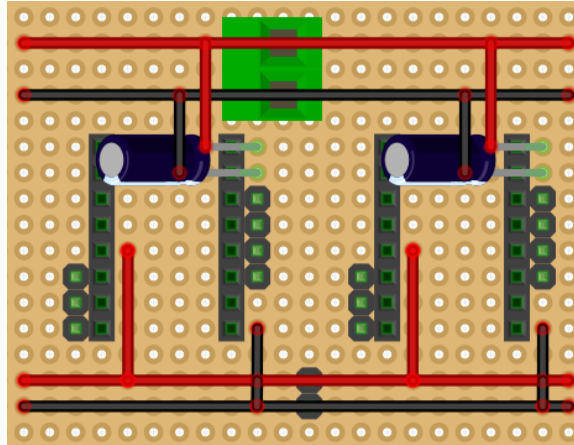


Figura 129. Placa on es connecten els drivers. (Font: Elaboració pròpia)

3.4.11 DISTRIBUÏDOR

El distribuïdor és simplement un conjunt de regletes connectades en sèrie per facilitar la distribució de l'alimentació de baixa potència a 5V des de la Raspberry Pi, ja que la placa només disposa de 2 pins GPIO de 5V i se'n necessiten 5.

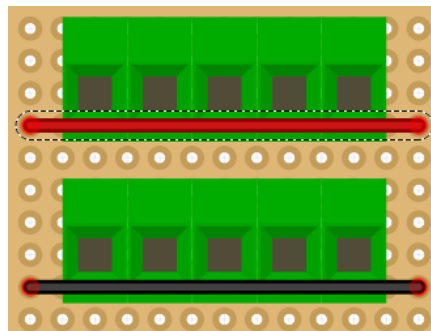


Figura 130. Esquema del distribuïdor. (Font: Elaboració pròpia)

3.5 PROGRAMA INFORMÀTIC

3.5.1 DIAGRAMA DE FLUX DEL COS DEL PROGRAMA

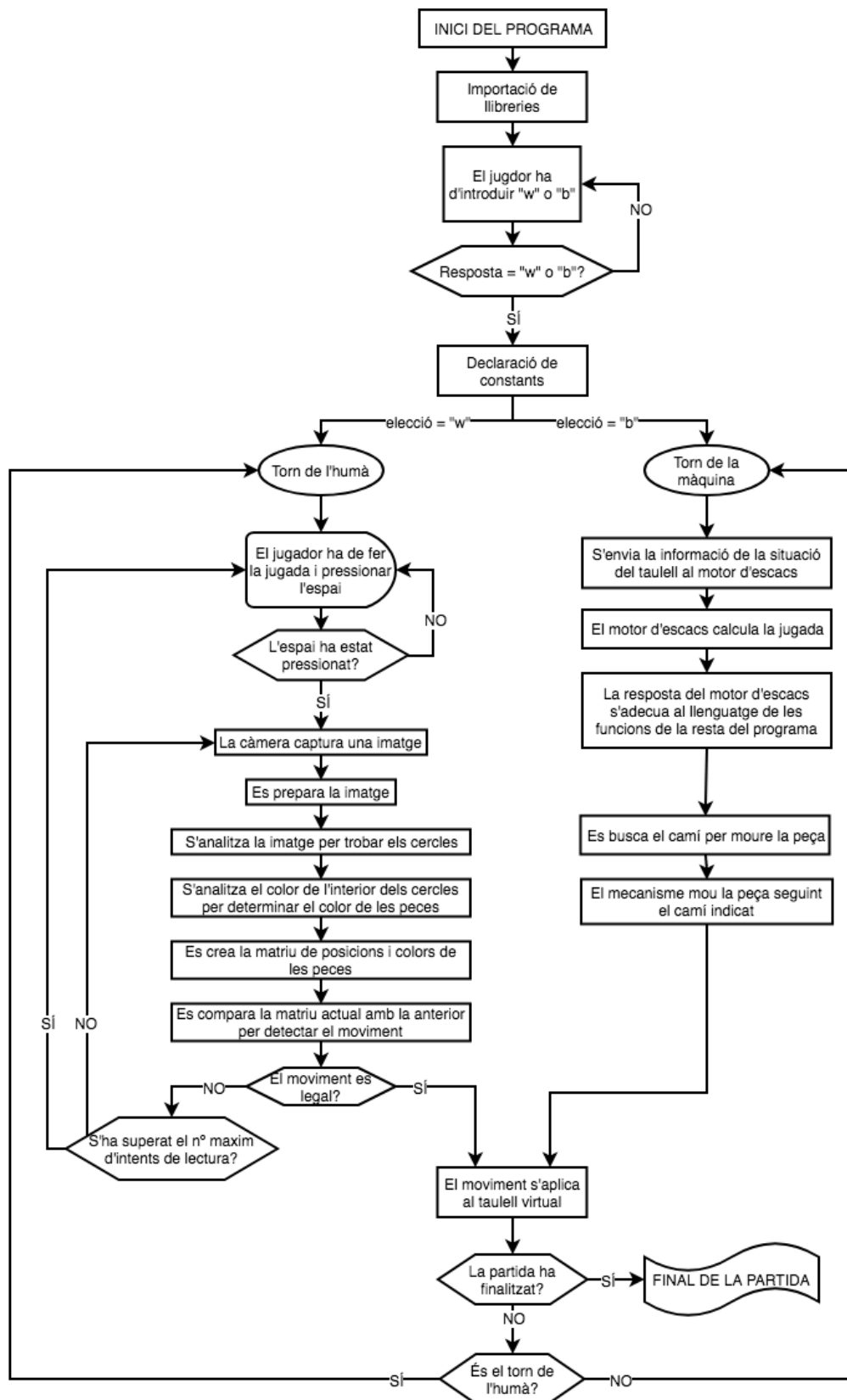


Figura 131. Diagrama de flux simplificat del cos del programa. (Font: Elaboració pròpia)

3.5.2 COS DEL PROGRAMA

En aquest apartat es farà un breu resum del procediment que segueix el programa en el seu funcionament.

NOTA 1: La numeració de les línies (dins els requadres) no tenen relació amb la numeració de les etapes del programa (fora dels requadres).

Llegenda de colors:

- Verd: Notes del programador. No afecten en el funcionament del programa, però ajuden a entendre'l per part d'un lector humà. (Si es vol entendre millor el programa recomano llegir-los).
- Blau fosc: Funcions bàsiques del programa, estructures principals del llenguatge de programació. (Per exemple importar, crear bucles, comparar, mostrar per pantalla, etc.)
- Blau clar: Text que es mostrarà per pantalla.
- Negre: La resta. (Exemple: Variables, operacions, valors, funcions, etc.)

```
1. import picamera # Llibreria Picamera, per treballar amb la càmera de la Raspber  
ry Pi  
2. import numpy as np # Llibreria Numpy, per treballar amb matrius en Python, nece  
saria per OpenCV  
3. import cv2 # Llibreria OpenCV, per anàlisi d'imatges  
4. import math # Llibreria d'operacions matemàtiques  
5. import chess # Llibreria Python-  
Chess, per a la generació i control del taulell d'escacs virtual  
6. import chess.uci # Llibreria per a la comunicació amb el motor d'escacs, en aqu  
est cas Stockfish. Aquest últim calcularà les jugades de la màquina  
7. import time # Llibreria per al control de temps  
8. #import pygame Llibreria per a la creació de jocs el Python
```

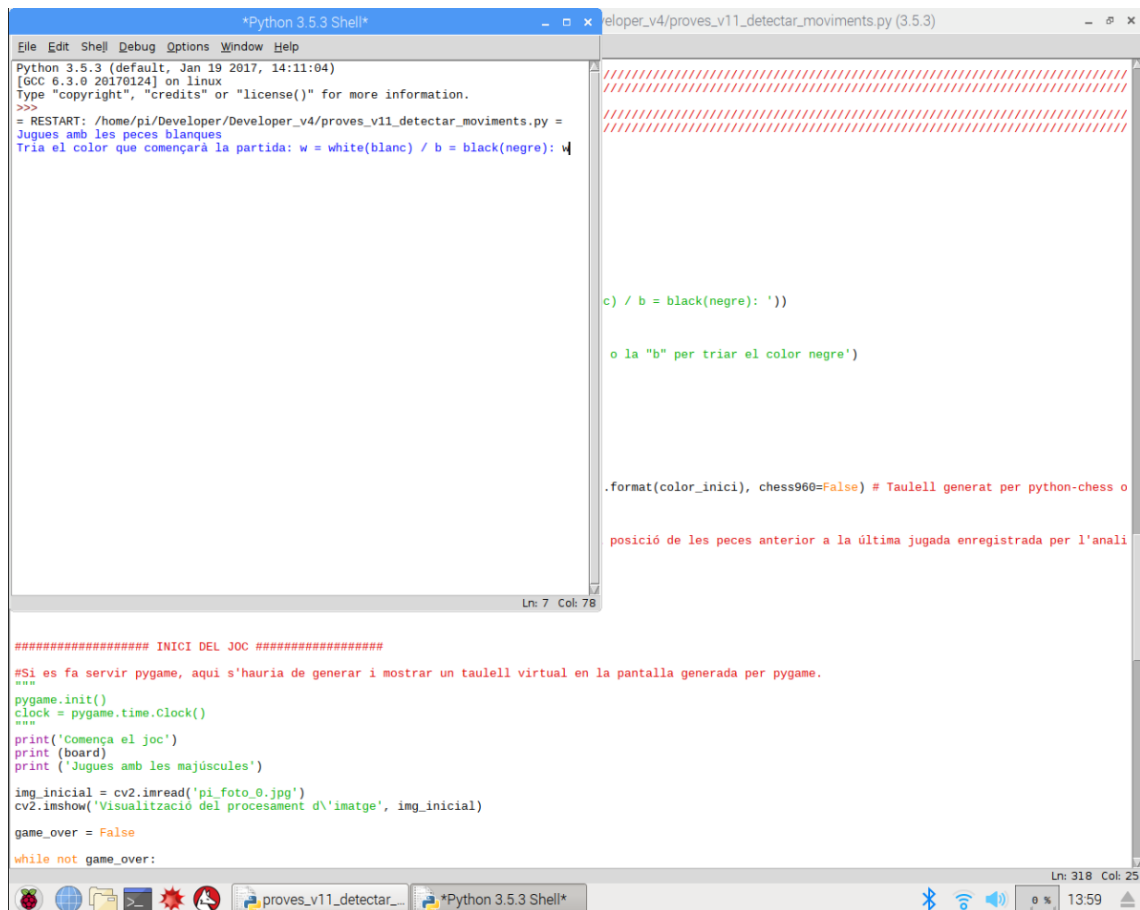
1. Inicialment s'importen les llibreries que contenen algunes de les funcions que s'utilitzaran en el programa.

```
1. ##### SELECTOR DE JUGADOR INICIAL #####  
2.  
3. while True:  
4.     print ('Jugues amb les peces blanques')  
5.     color_inici = str(input('Tria el color que començarà la partida: w = white(  
blanc) / b = black(negre): '))  
6.     if color_inici == 'w' or color_inici == 'b':  
7.         break  
8.     else:  
9.         print ('Entrada incorrecta, cliqueu la lletra "w" per triar el color bl  
anc, o la "b" per triar el color negre')
```

2. El programa s'inicia i indica a l'humà que juga amb les blanques.
3. Se l'hi permet a l'humà triar qui comença la partida.

Dispositiu automàtic per a jugar a escacs

Marçal Civill Cortada



```
*Python 3.5.3 Shell*
Python 3.5.3 (default, Jan 19 2017, 14:11:04)
[GCC 6.3.0 20170124] on linux
Type "copyright", "credits" or "license()" for more information.
>>>
= RESTART: /home/pi/Developer/Developer_v4/proves_v11_detectar_moviments.py =
Jugues amb les peces blanques
Tria el color que començarà la partida: w = white( blanc ) / b = black( negre ): w

c) / b = black( negre ): '))

o la "b" per triar el color negre')

.format( color_inici ), chess960=False) # Taulell generat per python-chess o
posició de les peces anterior a la última jugada enregistrada per l'anali

##### INICI DEL JOC #####

#Si es fa servir pygame, aquí s'hauria de generar i mostrar un taulell virtual en la pantalla generada per pygame.
pygame.init()
clock = pygame.time.Clock()
print('Comença el joc')
print( board )
print( 'Jugues amb les majúscules' )

img_inicial = cv2.imread( 'pi_foto_0.jpg' )
cv2.imshow( 'Visualització del procesament d\'imatge', img_inicial )

game_over = False

while not game_over:
```

Figura 132. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc. (Font elaboració pròpia)

Dispositiu automàtic per a jugar a escacs

Marçal Civill Cortada

```
1. ##### DECLARACIÓ DE CONSTANTS #####
2.
3. coord_encuadrar = np.array([153,14,1131,13,154,997,1149,985])
4. xmax = 800 #Resolució de la imatge encuadrada del taulell analitzada amb Open
   CV
5. amplada_escac = xmax/8 # Amplada (i altura) de cada un dels escacs (o caselles
   )
6.
7. camera = picamera.PiCamera()
8.
9. board = chess.Board(fen='rnbqkbnr/pppppppp/8/8/8/PPPPPPPP/RNBQKBNR {} KQkq -
   0 1'.format(color_inici), chess960=False) # Taulell generat per python-
   chess on es guardaran totes les dades de les peces
10.
11. t_ant = taulell_inicial_CV() # Taulell anterior utilitzat per la CV: emmagatzem
   a la posició de les peces anterior a la última jugada enregistrada per l'anàlis
   i d'imatge (info: blanc/negre)
12.
13. if color_inici == 'w':
14.     torn_huma = True # Emmagatzema un boleà en funció de si és o no torn de l'h
   umà
15. else:
16.     torn_huma = False
17.
18. max_intents = 3 # Limit d'intents de lectura en cas de detecció de jugada ilegala
19. centratge = 0.8 # Proporció mínima del radi de cada peça que ha d'estar dins de
   l'escac per ser detectada
20.
21. temps_calcul = 2000 # Temps de càlcul de la màquina (en milisegons). Com més te
   mps, més dificultat.
```

4. El programa segueix i en segon pla es declaren les constants que s'utilitzaran posteriorment en el programa. Estan totes agrupades per facilitar la seva modificació.

```
1. ##### INICI DEL JOC #####
2.
3. #Si es fa servir pygame, aquí s'hauria de generar i mostrar un taulell virtual
   en la pantalla generada per pygame.
4.
5. print('Comença el joc')
6. print ('Les majúscules son les blanques')
7. print ('#####')
8. print (board)
9. print ('#####')
10.
11. img_inicial = cv2.imread('pi_foto_0.jpg')
12. cv2.imshow('Visualització del procesament d\'imatge', img_inicial)
13.
14. engine = chess.uci.popen_engine("/usr/games/stockfish", setpgroup=True) # Ruta d'
   accés a l'arxiu executable de Stockfish
15. engine.uci() # Inici de la comunicació amb Stockfish
16.
17. game_over = False
```

5. Per pantalla se l'hi indica al jugador que el joc ha començat.
6. Es mostra el taulell virtual inicial.
7. Es mostra una foto del taulell físic inicial.
8. S'inicia la comunicació amb el motor d'escacs (el que calcula les jugades).

Dispositiu automàtic per a jugar a escacs

Marçal Civill Cortada

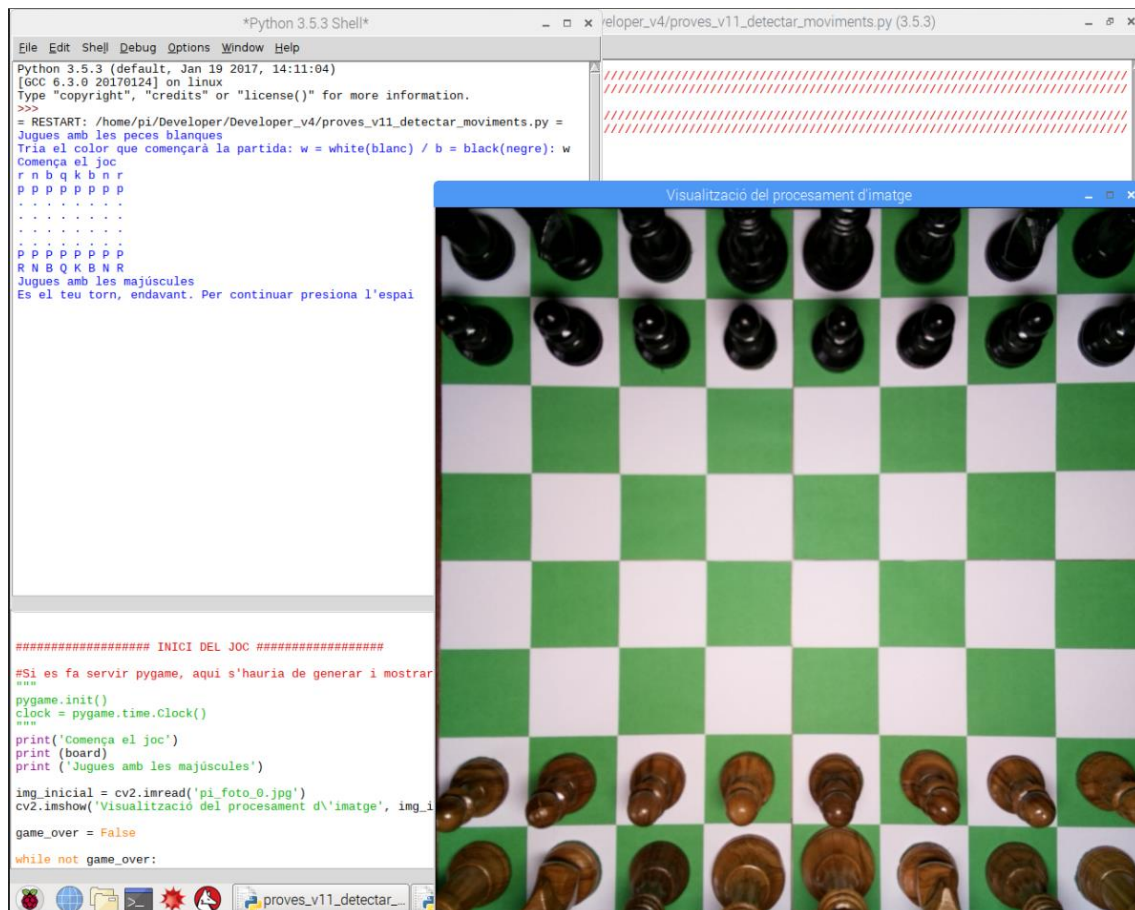


Figura 133. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc. (Font elaboració pròpia)

```
1. while not game_over:
2.
3.     if torn_huma:
4.
5.         print ('Es el teu torn.')
6.
7.         jugada_ilegal = True
8.
9.         while jugada_ilegal: #Comença el bucle d'analisi d'imatge i comprovació
            de jugades vàlides. Es sortirà del bucle quan l'humà faci una jugada vàlida.
10.
11.             print ('Esperant la teva jugada. Per continuar presiona l\'espai')
12.             cv2.waitKey(0)
```

9. Aquí comença el bucle del joc del qual no sortirà fins que s'acabi la partida o es tanqui el programa.
10. Comprova si és el torn de l'humà, si ho és ho indica per pantalla i espera que l'humà faci la seva jugada i pressioni l'espai.

```
1. ##### L'HUMÀ HA MOGUT, COMENÇA L'ANÀLISI D'IMATGE #####
2.
3.         lectura_erronia = True
4.         intents_lectura = 1
5.
6.         while lectura_erronia:
7.
8.             camera.capture('pi_foto.jpg') # Es captura una imatge amb la cà
           mara i es guarda
9.             foto = cv2.imread('pi_foto.jpg') # Es llegeix la imatge guardad
           a per la càmera
10.
11.             img_enc = encuadrar(foto, coord_encuadrar) # S'encuadra l'area d
           e la foto que conté el taulell
12.
13.             img_180 = r180(img_enc) # Es gira la imatge 180º perquè la càma
           ra està orientada en sentit contrari al jugador humà.
14.             cv2.imshow('Visualització del procesament d\'imatge', img_180)
           # Es mostra la imatge
15.             cv2.waitKey(1000) # S'espera 1 segon per poder visualitzar la i
           matge
```

11. Quan l'humà pressiona l'espai comença l'anàlisi d'imatge que es repetirà fins que la lectura sigui correcta o fins que es superin el nombre d'intents de lectura límit.
12. La càmera captura la imatge i la guarda.
13. Es llegeix la imatge, i s'activen les funcions per enquadrar-la i girar-la.
14. Finalment la imatge es mostra per pantalla durant un segon.

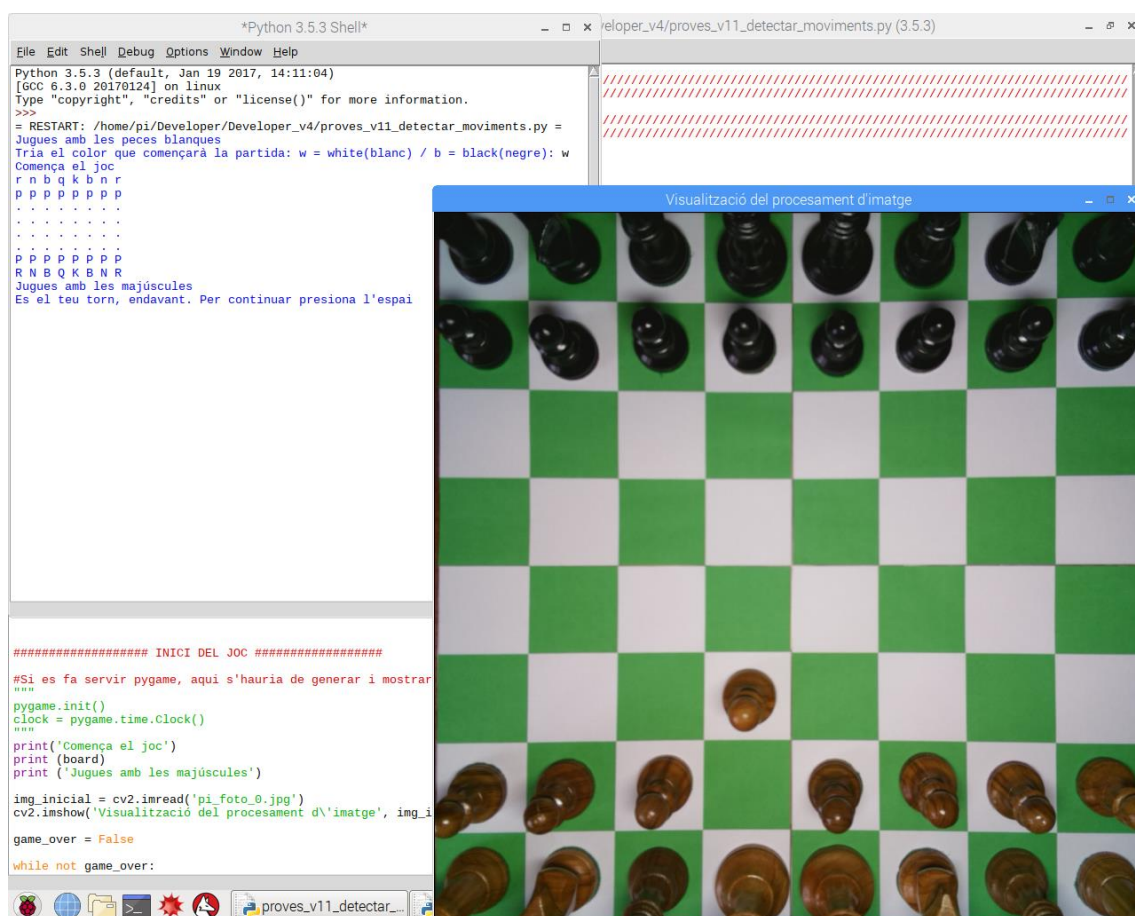


Figura 134. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc.

(Font elaboració pròpia)

```
1.         cercles_llista, cercles_img = cercles(img_180) # Es detecten el
           s cercles i es guarden, també es dibuixen sobre la imatge.
2.         print('cercles:')
3.         print(cercles_llista)
4.         cv2.imshow('Visualització del procesament d\'imatge', cercles_i
           mg)
5.         cv2.waitKey(1000)
```

15. S'agafa la imatge i s'envia a través de la funció que detectarà en quins escacs hi ha peces (veure 3.5.5. *Funció per detectar les peces* d'aquest document). Aquesta informació la guardarà en una llista ordenada anomenada `cercles_llista`.
16. Es mostra una imatge amb els cercles detectats i validats dibuixats sobre la foto del taulell juntament amb la llista de cercles.
17. S'espera a que el jugador pressioni l'espai per continuar.

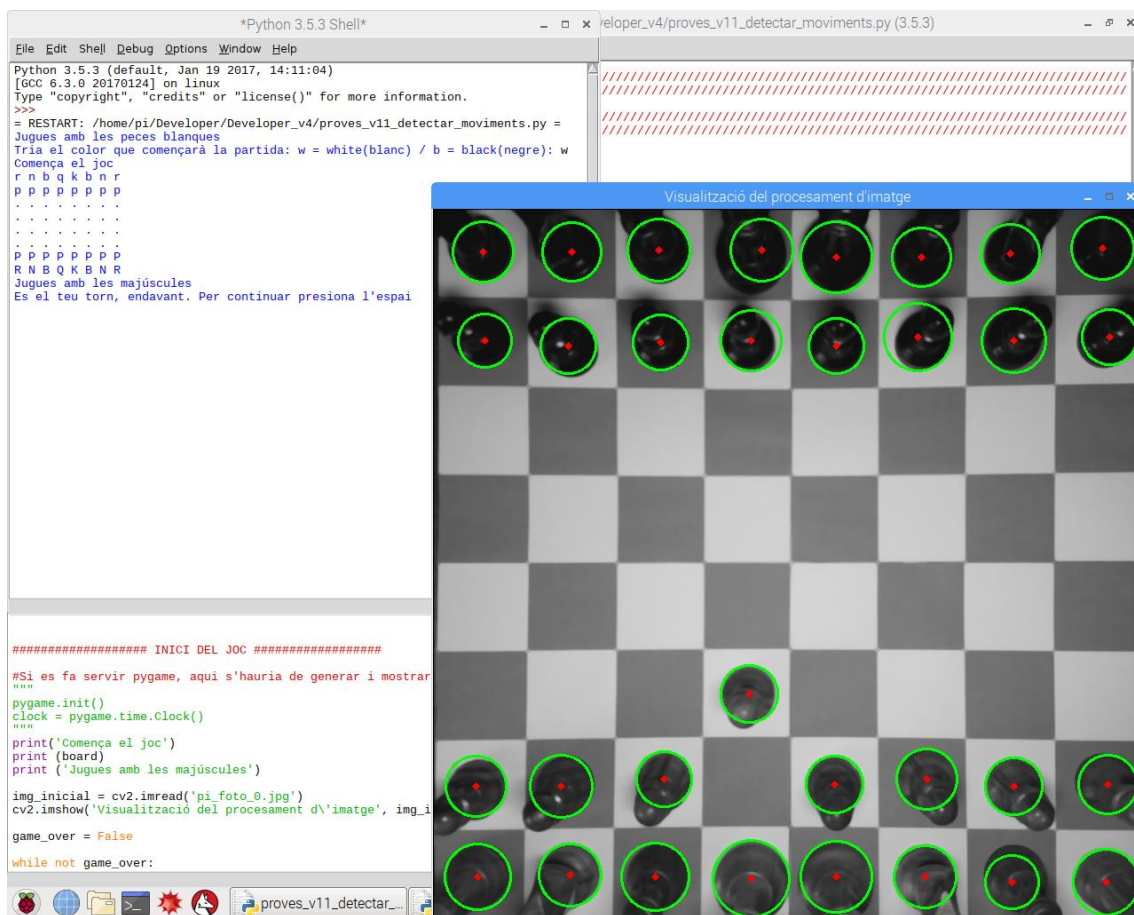


Figura 135. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc.
(Font elaboració pròpia)

Dispositiu automàtic per a jugar a escacs

Marçal Civill Cortada

```
1.         pix_cons_img = np.copy(cercles_img)
2.
3.         P_BN_llista, pix_cons_img = posicio_color(img_180, pix_cons_img
, cercles_llista)
4.         print('P_BN_llista:')
5.         print(P_BN_llista)
6.         cv2.imshow('Visualització del procesament d\'imatge', pix_cons_
img)
7.         cv2.waitKey(1000)
```

18. La llista amb les posicions de les peces detectades juntament amb la imatge del taulell es passa a la funció encarregada de detectar el color de cada una de les peces detectades.

19. Es mostra una imatge representativa del procediment utilitzat per detectar el color juntament amb la llista de les posicions de les peces amb el seu corresponent color (Blanc/Negre).

20. S'espera a que el jugador pressioni l'espai per continuar.

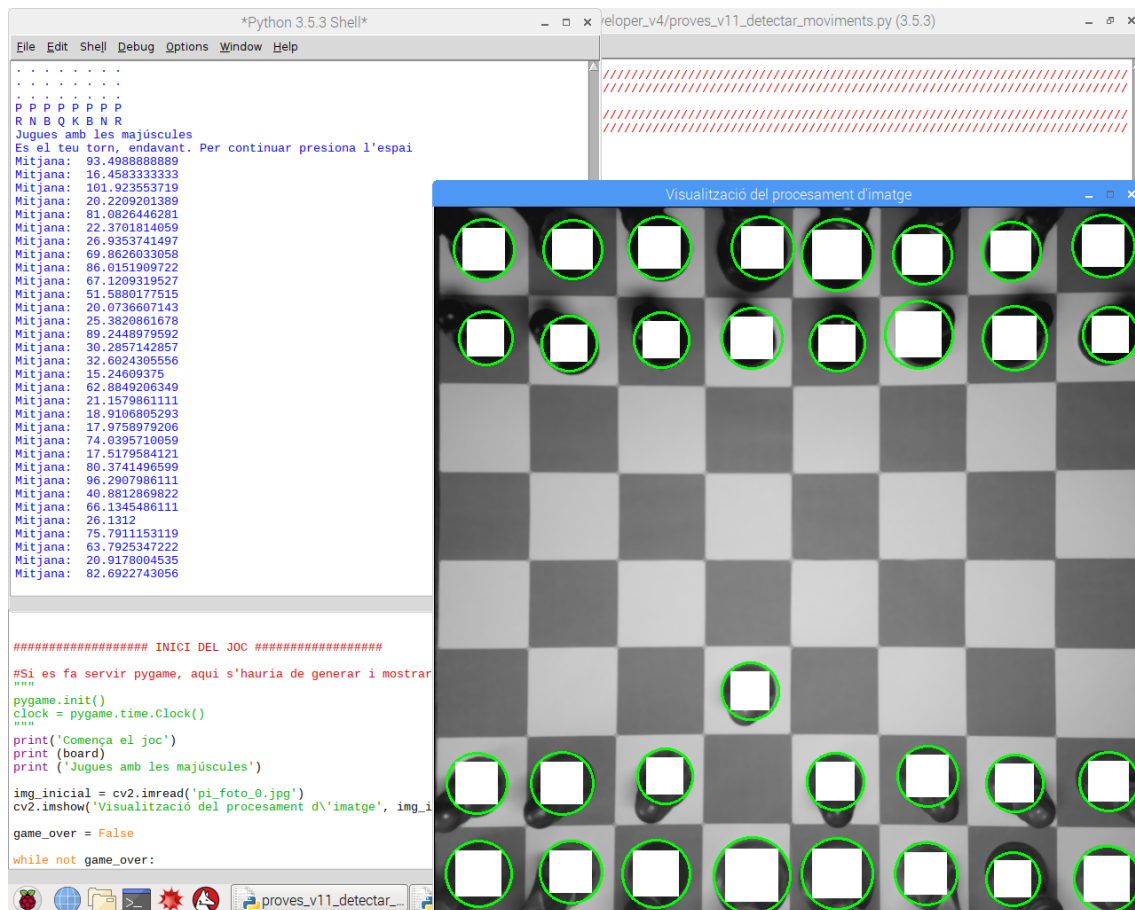


Figura 136. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc. (Font elaboració pròpia)

Dispositiu automàtic per a jugar a escacs Marçal Civill Cortada

```
1. tau_CV = np.zeros((8,8), dtype=int) # Es crea una matriu 8x8 plena de "0" que representa un taulell buit
2.
3. for i in P_BN_llista: # Es crea una matriu de posicions i colors a partir de les posicions i colors guardats en P_BN_llista
4.     tau_CV[i['py']][i['px']] = i['c']
5.
6. print ('tau_CV:')
7. print (tau_CV)
8.
9. mov = moviments_detectats_CV(t_ant, tau_CV)
10. print ('La teva jugada:', mov)
```

21. Es crea una matriu 8x8 on es representen els escacs buits, els ocupats per peces blanques i els ocupats per peces negres.
22. Es mostra la matriu de posicions per pantalla.
23. S'activa la funció per detectar els moviments. Els moviments es detecten comparant la matriu creada en aquest moment amb la creada en el torn anterior.
24. Es mostra per pantalla el moviment detectat.

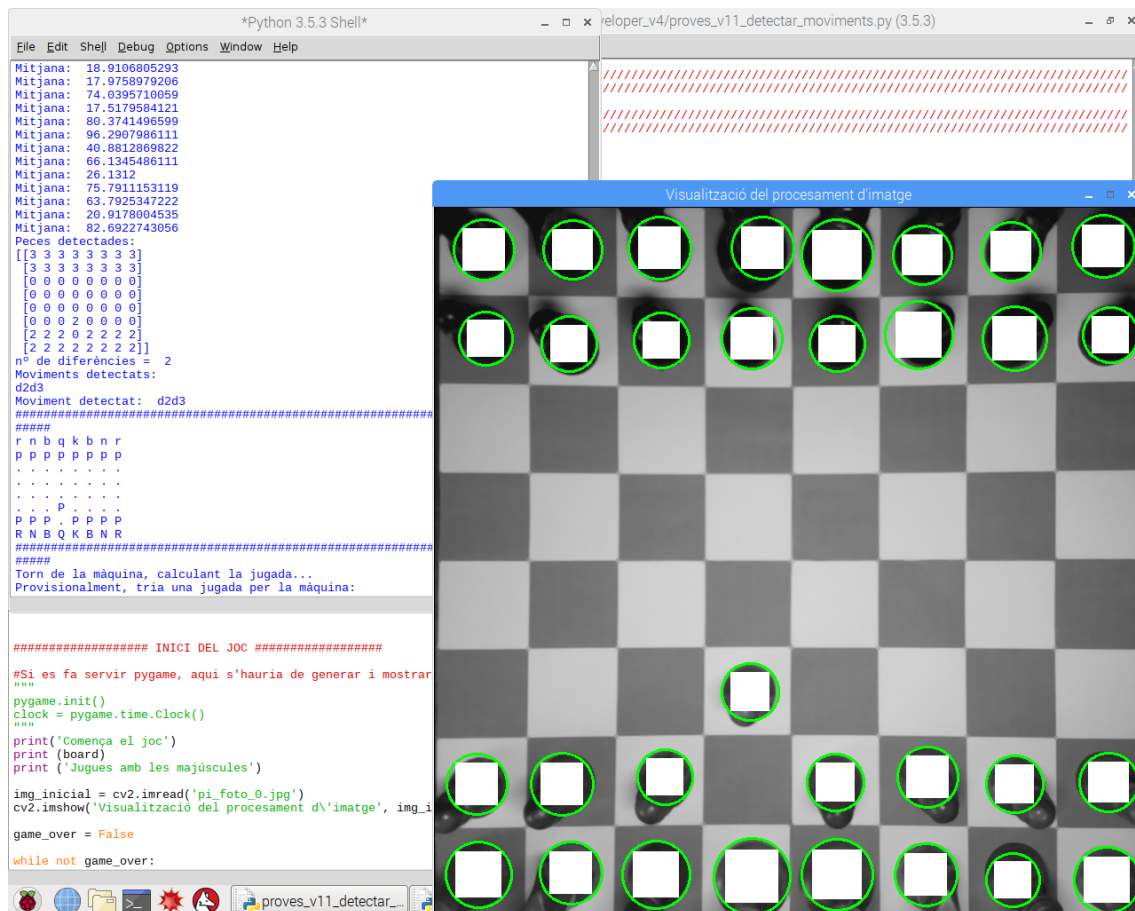


Figura 137. Captura del que pot veure el jugador mostrant-se per pantalla en aquesta etapa del joc. (Font elaboració pròpia)


```
1.         if (mov in board.legal_moves) or (mov in board.pseudo_legal_moves): # Es comprova si el moviment es legal o pseudo-legal
2.             if board.is_into_check(mov): # Comprova si s'està deixant el propi rei en escac
3.                 print('Estas deixant el rei en escac')
4.
5.                 lectura_erronia = False # Es surt del bucle de lectura
6.                 pero es continua en el bucle de jugada humana
7.                 print('Moviment il·legal. A la espera d\'una nova jugada')
8.             else:
9.                 lectura_erronia = False # Es surt del bucle de lectura
10.                jugada_ilegal = False # I també es surt del bucle de jugada humana
11.
12.                elif intents_lectura < max_intents: # Comprova si el nombre d'intents de lectura ha arribat al límit establert (apartat "declaració de constants")
13.                    intents_lectura += 1
14.                    print('Reintentant la lectura per {}a vegada...'.format(intents_lectura))
```

25. Es comprova que el moviment detectat sigui legal. Si ho és s'espera que el jugador faci una jugada vàlida.

```
1.         else:
2.             lectura_erronia = False
3.             print('Moviment il·legal. A la espera d\'una nova jugada')
4.
5.             print ('Nº intents de lectura: ', intents_lectura)
6.             t_ant = np.copy(tau_CV) # El taulell actual es guarda com a anterior per al pròxim anàlisi
7.
8.             torn_huma = False
9.
10.            ##### ANALISI D'IMATGE FINALITZADA, MOVIMENT DETECTAT I TAULELL ACTUALITZAT, TORN HUMÀ FINALITZAT #####
```

26. Si la jugada es legal es surt del bucle.

27. Es guarda una còpia de les posicions en el taulell per comparar-lo en la següent jugada.

28. S'indica que el torn de l'humà ha finalitzat.

```
1.     else:
2.         print ('Torn de la màquina, calculant la jugada...')
3.
4.         engine.position(board, async_callback=None) # S'envia la informació de
la situació del taulell a Stockfish
5.         bestmove, ponder = engine.go(movetime=temps_calcul) # Stockfish calcula
la jugada i l'assigna a la variable "best"
6.
7.         mov = str(bestmove)
8.         print ('Jugada de la màquina: ', mov)
9.
10.        print ('Mou la peça per la màquina i presiona l\'espai')
11.        cv2.waitKey(0) # Pausa provisional per moure les peces manualment per l
a màquina
```

29. Es mostra per pantalla que és el torn de la màquina.

30. La informació del taulell virtual de Python-Chess s'envia al motor d'escacs Stockfish perquè calculi la jugada de la màquina.

31. La jugada de la màquina es mostra per pantalla.

```
1.         # comprova len(mov) per si te una q al final i eliminar-
la (en una copia. el "mov" ha de quedar amb la q)
2.
3.         if len(mov) == 4:
4.             a,b,c,d = list(mov) # Es descomposa la jugada de la màquina per pod
er actualitzar el t_ant
5.         else: # El moviment porta una 5a lletra que indica la promoció d'un peó
6.
7.             a,b,c,d,e = list(mov)
8.
9.             a = a.upper()
c = c.upper()
```

32. Es condiona la resposta del motor d'escacs per adaptar-la al funcionament del programa.

```
1.         pi = chess.square_mirror(eval('chess.'+ a + b)) # eval() function conve
rts string to executable code
2.         fila_pi = chess.square_rank(pi)
3.         columna_pi = chess.square_file(pi)
4.
5.         pf = chess.square_mirror(eval('chess.'+ c + d))
6.         fila_pf = chess.square_rank(pf)
7.         columna_pf = chess.square_file(pf)
8.
9.         t_ant[fila_pi][columna_pi]=0
10.        t_ant[fila_pf][columna_pf]=3
```

33. Es converteix la resposta del motor d'escacs en els valors adequats per modificar la matriu de posicions que emmagatzema les posicions de les peces.

```
1.         if board.is_kingside_castling(chess.Move.from_uci(mov)): # Si es un enr
oc, actualitza també la posició de la torre. La posició del rei ja s'ha actuali
tzat.
2.             t_ant[0][7]=0
3.             t_ant[0][5]=3
4.         elif board.is_queenside_castling(chess.Move.from_uci(mov)):
5.             t_ant[0][0]=0
6.             t_ant[0][3]=3
7.         elif board.is_en_passant(chess.Move.from_uci(mov)): # Es comprova si el
moviment és una captura al pas del peó.
8.             t_ant[fila_pf-
1][columna_pf]=0 # S'elimina el peó blanc capturat al pas (que estarà situat en
la mateixa columna, fila posterior)
9.
10.        mov = chess.Move.from_uci(mov) # El moviment es passa al llenguatge del
"board"
11.
12.        ##### Etapa de camins i motors #####
```

34. Es comprova si la màquina ha decidit fer un dels moviments especials que requereixen un processament addicional per actualitzar la matriu de posicions:
35. Es comprova si el moviment de la màquina és un enroc, si ho és s'actualitza també la posició de la torre en la matriu de posicions.
36. Es comprova si el moviment de la màquina és una captura al pas del peó, si ho és s'esborra de la matriu el peó eliminat.
37. Finalment la informació del moviment s'adequa al llenguatge del taulell virtual perquè pugui ser llegida.

```
1.         espais_TF = espais(P_BN_llista, fila_pi, columna_pi, fila_pf, columna_p
f) # (espais_TF = Espais True/False)
2.         print(espais_TF)
3.
4.         #####
5.
6.         camera.capture('pi_foto.jpg')
7.         foto = cv2.imread('pi_foto.jpg')
8.         img_enc = encuadrar(foto, coord_encuadrar)
9.         img_180 = r180(img_enc)
10.        cv2.imshow('Visualització del procesament d\'imatge', img_180)
11.
12.        torn_huma = True
```

38. S'activa la funció que genera el mapa per al cercador de camins. Aquest mapa indica quines posicions estan ocupades i quines estan lliures per al pas.
39. Pendent de programar: El buscador de camins busca la ruta més curta per moure la peça de la màquina.
40. Pendent de programar: Els motors segueixen la ruta per portar la peça de la posició inicial a la final.
41. Es crea una foto del taulell que es mostra per pantalla per al següent torn i es dona per finalitzat el torn de la màquina.

```
1. #////////////////////////////////////  
2. #Apartat comú d'humà i màquina  
3. #////////////////////////////////////  
4.  
5. board.push(mov)  
6. print ('#####')  
7. print (board)  
8. print ('#####')
```

42. Un cop s'ha acabat un torn, sigui d'humà o de màquina, la posició detectada es passa a la funció que actualitza el taulell virtual.

43. El taulell virtual es mostra per pantalla.

```
1. game_over = board.is_game_over(claim_draw=False)  
2. if game_over:  
3.     print ('#####')  
4.     print ('#####')  
5.  
6.     print ('FINAL DE LA PARTIDA')  
7.  
8.     if board.is_checkmate():  
9.         print('Situació final: Escac i mat')  
10.  
11.     if board.is_stalemate():  
12.         print('Situació final: Taules')  
13.     else:  
14.         if board.turn:  
15.             print('Ha guanyat la màquina')  
16.         else:  
17.             print('Has guanyat!')  
18.  
19.     print ('#####')  
20.     print ('#####')  
21.  
22. engine.quit()
```

44. Es comprova si el joc ha acabat. Si és així es comprova el motiu i es mostra per pantalla el missatge “FINAL DE LA PARTIDA”, el guanyador i la situació final de la partida.

4. CONCLUSIONS

Anàlisi d'imatge:

Per a l'anàlisi d'imatge és fonamental una bona il·luminació, sembla obvi però una bona il·luminació significa una il·luminació directa, sense reflexos i sense ombres.

Els colors del fons han de generar un bon contrast amb el color dels objectes que es volen detectar.

Els colors del fons han de ser clors llisos i homogenis per evitar les falses deteccions.

Són molt més importants els punts anteriors que una càmera de bona qualitat.

Abans de fer les proves d'imatge decideix quines peces exactes vols detectar i sobre quin fons. Quan ho sàpigues comença a fer les proves d'anàlisi d'imatge, canviar la mida o color de les peces després d'haver acabat el programa d'anàlisi pot suposar moltes hores per tornar-lo a calibrar.

Disseny mecànic:

Abans de dissenyar una peça busca si ja hi ha al mercat una peça similar que et pugui servir. T'estalviaràs molta feina i probablement diners.

La simplicitat en el disseny és la clau, com més complex sigui més coses poden sortir malament i més difícil serà fabricar i muntar la màquina.

Tingues clara la funció de cada peça per determinar la precisió que necessites en cada cota. No fabriquis amb més precisió de la necessària o et gastaràs diners inútilment.

No cal comprar peces de precisió si la màquina no ha de fer una feina de precisió, i pensa que sempre pots ajustar la posició de les peces en el moment de muntar-les si l'error no és massa gran i la màquina no requereix d'una extrema precisió.

La fusta contraxapada és molt resistent i molt lleugera.

Moltes de les rodes i politges barates del mercat porten rodaments interiors, però no porten casquillos entre ells i si apretes el coixinet al fer el muntatge amb un cargol aquests es bloquegen completament. Al fer el muntatge amb aquest tipus de components s'ha de deixar una mínima separació entre la roda o politja i els elements que la subjecten lateralment.

Utilitzar un full de càlcul per a les cotes pot semblar laboriós però pot estalviar molta feina si es detecta un error de càlcul, ja que totes les cotes encadenades es re-calcularan automàticament.

Abans de fabricar tingues tota la màquina mutada en un 3D i comprova que no hi ha col·lisions. Començar a fabricar el prototip quan encara hi ha peces per acabar de dissenyar provocarà que amb força seguretat et trobis amb problemes de muntatge i hakis de repetir el prototip.

Inclou els dispositius elèctrics i electrònics en el disseny 3D. No cal que estiguin dibuixats amb precisió però si que representin l'espai que ocuparan.

La fusta contraxapada prima (de 5mm) tendeix a corbar-se per si mateixa, s'ha de tenir en compte al dissenyar un taulell que sigui perfectament pla.

Els vinils s'enganxen perfectament a la superfície de la fusta.

Si alguna cosa es pot provar físicament i fàcilment prova-la i acabaràs abans que fent un munt de suposicions i deixaràs la ment lliure per decisions més importants.

Programació informàtica:

El motor d'escacs Stockfish et pot guanyar sense problemes amb un segon de càlcul a no ser que siguis un expert jugador d'escacs.

Tot el que vulguis saber per iniciar-te i formar-te fins a un bon nivell en la programació ho pots trobar a internet. Però aprendre a programar sense que algú te n'ensenyi és molt complicat i porta molt de temps, a part probablement faràs servir mètodes poc eficients en els teus programes. El millor és aprendre a programar amb un professor i després aprendre pel teu compte i quan hakis practicat molt, si t'interessa saber més torna a estudiar amb un professor i així tantes vegades com et vingui de gust. La informació hi és, però estructurar-la d'una manera eficient i amb una certa metodologia i eficiència és molt difícil sense que algú t'ho ensenyi. El mateix passa amb les altres àrees del coneixement, però comparada amb les altres, la informació sobre programació que hi ha a la xarxa és descomunal.

Hi ha una gran comunitat que comparteix la filosofia Open Source i pots trobar llibreries gratuïtes de tants tipus i amb tantes funcions cada una que o te les pots acabar.

Electrònica

Una Raspberry Pi pot fer funcionar un programa d'escacs sense problema i a una gran velocitat.

La Raspberry Pi es pot utilitzar com un ordinador, pots utilitzar editors de textos, navegar per internet, etc. Però si obres masses programes a la vegada s'acostuma a penjar a causa de la seva poca RAM (1GB).

Raspberry Pi i Arduino són dues plataformes molt bones per iniciar-se en la robòtica, però Raspberry ofereix les característiques perfectes per a aquest projecte.

La Raspberry Pi no pot llegir entrades analògiques però es pot solucionar amb un microcontrolador entre l'entrada analògica i la Raspberry.

Fabricació:

En una fusteria o una taller d'alumini la precisió que utilitzen és del voltant d'1mm. Si vols més precisió tens dues opcions, o anar a un centre de mecanitzat o anar tallant per prova i error en un taller fins a obtenir la mida necessària.

Compta que les hores de taller estan per sobre els 20€/h, el material és molt més barat que les hores de feina. A comparació del preu de les hores de feina el material surt pràcticament gratuït.

El que puguis fabricar tu amb la precisió suficient fes-ho, t'estalviaràs un munt de diners, sobretot si és el primer prototip. Si un cop fet el prototip es veu que es necessita més precisió ja aniràs a un centre de mecanitzat.

Gestió d'un projecte nou i autogestionat a llarg termini:

Un projecte d'aquest tipus és una cursa de resistència, no un esprint, no vulguis córrer o t'acabaràs cremant i no acabaràs.

L'ansietat i l'estrès són un factor molt important que es pot suportar en èpoques curtes com per exemple èpoques d'exàmens, però en un projecte que dura tant de temps si no controles l'estrès aquest t'acabarà destrossant física i mentalment.

Respecta els teus horaris de son i les pauses per al descans. Compleix un mínim horari de (hora de llevar-te i d'anar a dormir). No intentis acabar un disseny si t'estàs passant

de l'hora d'anar a dormir, l'endemà el projecte seguirà allà, apunta totes les idees que tinguis en un full i vés a dormir. L'endemà lleva't a l'hora que tinguis establerta i segueix. Mentalment rendiràs molt més si has dormit que si tens son encara que quan estàs amb totes les idees voltant pel cap a les tantes de la nit creguis que ho fas llavors o te n'oblidaràs.

No deixis de fer esport mentre duri el projecte. Quan vegis que no avances tant ràpid com esperaves voldràs eliminar coses que fas i que et treuen temps. No eliminis l'esport, és molt important per al bon funcionament mental i per al control de l'estrès a part de la resta de la salut física. Segur que pots deixar de perdre temps en moltes altres coses.

En la mateixa línia dels horaris, compleix un horari alimentari i fes tots els àpats, si comences a deixar-te guiar per les emocions i les ganes d'acabar coses en el projecte i comences a desquadrar horaris de son o d'alimentació acabaràs molt perjudicat.

Fes planificacions generals d'objectius específics cada dos o tres dies sense invertir-hi molt de temps (sense horari ni calendari, només una llista de tasques). Probablement aquests objectius no els realitzaràs en ordre, alguns els modificaràs, altres els eliminaràs pel camí i te n'apareixeran de nous. Simplement vés completant les tasques de la llista sense pressa però sense pausa i renovant la llista quan ho consideris necessari.

Apunta en un full totes les idees que se t'acudeixin, no intentis guardar-les totes al cap o se t'oblidaran o be et provocaran insomni.

És bo tenir un calendari general per assegurar-te de no dedicar més temps a una àrea que a una altra i a ajudar-te a controlar els terminis.

És important que en les planificacions temporals afegeixis un bon marge de temps per imprevistos i amb un bon marge de temps em refereixo a que reservis com a mínim el doble de temps del que tens previst.

És important avançar de la forma homogènia en tots els fronts del disseny, no consumeixis molt de temps perfeccionant una part si en tens una altra que encara està en una fase de disseny inicial. Si ho fas probablement quan avancis la part que ha quedat enrere t'obligarà a fer modificacions en altres àrees connectades, a les que prèviament hauràs estat invertint molt de temps per perfeccionar-les i aquest temps haurà estat en va.

5. LÍNIES FUTURES DE TREBALL

- Acabar les funcions que falten.
- Estructuració del programa en blocs més compactes.
- Millora de estabilitat de la estructura vertical per reduir les vibracions de la càmera.
- Millora de la organització del cablejat.
- Disseny de guies cilíndriques mòbils en comptes de guia quadrada mòbil.
- Disseny de sistema d'engranatges multiplicadors per a motors pa a pas.
- Detecció directa de les peces amb visió artificial.
 - Afegir més càmeres des d'altres angles.
 - Increment de la robustesa de la detecció, per detectar peces de diferents mides, formes i colors.
- Redisseny del sistema porta cables fins al servomotor.
- Incorporar botons i LED per poder eliminar la pantalla, teclat i ratolí.
- Fer càlculs estàtics i dinàmics i proves dinàmiques i redissenyar en conseqüència si és necessari.

PRESSUPOST

COMPONENTS DE MECÀNICA			
DESCRIPCIÓ	QUANTITAT	PREU	SUBTOTAL
Arandela M3	20	0,00 €	0,04 €
Arandela M8	4	0,01 €	0,04 €
Arandela M4 ample	130	0,01 €	0,78 €
Tub quadrat d'alumini 20x20 1m	1	6,95 €	6,95 €
Brides 2,5x100	1	1,44 €	1,44 €
Cinta adhesiva de doble cara	1	2,50 €	2,50 €
Escaire de 30x30 d'extrems arrodonits	24	0,08 €	1,82 €
Escaire d'unió 40x40x40x2	2	0,38 €	0,76 €
Escaire d'unió 50x50x15x2	2	0,40 €	0,80 €
Escaire d'unió 50x50x40x2	1	0,52 €	0,52 €
Grapa per tub rodó de 6mm	4	0,05 €	0,22 €
Tub rodó d'alumini anoditzat 1m	2	2,47 €	4,93 €
REMACHE MAQUINA NORMAL 4-10	4	0,03 €	0,10 €
Rebló 5x10 per reblonadora manual	14	0,04 €	0,50 €
Rebló roscat M4	4	0,12 €	0,47 €
Inserts fusta-ferro M4	8	0,38 €	3,04 €
TACO SILLA PLAST. ROSCADO 20x20	2	0,21 €	0,42 €
Tap rectangular de plàstic 40x20	5	0,11 €	0,56 €
Felpe	1	0,60 €	0,60 €
Potes de plastic	8	0,04 €	0,31 €
Cargol de fusta de 4x16	120	0,03 €	4,14 €
Cargols M2 de longitud única	4	0,02 €	0,08 €
Cargols M3 de diversa longitud	20	0,03 €	0,52 €
Cargols M4 de diversa longitud	10	0,03 €	0,34 €
Cargols M8x40	2	0,10 €	0,20 €
Femella M2	12	0,01 €	0,12 €
Femella M3	30	0,01 €	0,26 €
Femella M4	20	0,01 €	0,12 €
Femella M8	4	0,02 €	0,09 €
Femella autoblocant M3	10	0,01 €	0,11 €
Femella autoblocant M4	10	0,01 €	0,09 €
Peces d'escacs	1	10,90 €	10,90 €
Engranatges conics	4	2,40 €	9,60 €
Politges motor GT2 60 dents	2	10,50 €	21,00 €
Rodes V-Slot	8	2,26 €	18,08 €
Politges lliures dentades	6	2,10 €	12,60 €
Politges lliures planes	2	2,10 €	4,20 €
Corretja GT2 3m	2	6,45 €	12,90 €
Tensor corretja	2	0,90 €	1,80 €
Motor pas a pas NEMA 17	2	11,49 €	22,98 €
SUBTOTAL MECÀNICA			146,84 €

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

COMPONENTS D'ELECTRÒNICA			
DESCRIPCIÓ	QUANTITAT	PREU	SUBTOTAL
Alimentador universal 12vdc 3a 36w	1	9,25 €	9,25 €
BASE EMPORTAR CONECTOR alimentació	1	0,41 €	0,41 €
Cable flexible 0,25mm bossa 10mts negre	1	2,14 €	2,14 €
Cable flexible 0,25mm bossa 10mts vermell	1	2,14 €	2,14 €
Cable flexible 0,75mm bossa 10mts groc	1	3,45 €	3,45 €
Cable flexible 0,75mm bossa 10mts violeta	1	3,45 €	3,45 €
CABLE PARAL·LEL VERMELL I NEGRE 2x0,75	3	0,28 €	0,83 €
CABLE TELEFONIC C/ F 6x0. 22	1,5	0,28 €	0,42 €
Cconnexió red cr 80	1	1,41 €	1,41 €
Condensador elec. 100uf 16 v	2	0,08 €	0,16 €
Endoll doble DE SUPERFICIE	1	3,15 €	3,15 €
Estany fil sn60pb38cu2 1mm 100gr	1	4,42 €	4,42 €
Interruptor basculant amb llum	1	1,69 €	1,69 €
PLACA BAQUELITA TIRAS 100x160	1	5,41 €	5,41 €
Regleta 40c recte senc. Femella	3	0,55 €	1,65 €
Regleta 40c recte senc. Mascle	2	0,36 €	0,72 €
Regleta ak100/2	3	0,24 €	0,72 €
Regleta ak100/3	2	0,21 €	0,43 €
Regleta connexio 4mm 12 ele	1	0,51 €	0,51 €
Tubo retractil 2.4mm negre/vermell	1	0,60 €	0,60 €
Tubo retractil 4.8 negre/vermell	3	0,76 €	2,28 €
Tubo retractil 4mm negre/vermell	2	0,64 €	1,28 €
VENTILADOR 50x50x10 12V	2	2,96 €	5,92 €
Tub led	2	15,00 €	30,00 €
Potenciòmetres 10 voltes 10k	2	10,50 €	21,00 €
Targeta SD	1	11,99 €	11,99 €
Cable càmera 1m	1	6,49 €	6,49 €
Raspberry Pi càmera module V2	1	25,90 €	25,90 €
Raspberry pi 3 model b	1	36,26 €	36,26 €
Font alimentació 5V 3A	1	8,09 €	8,09 €
Teclat + ratolí	1	19,37 €	19,37 €
Driver DRV8825	2	4,69 €	9,38 €
SUBTOTAL ELECTRÒNICA			211,66 €

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

FABRICACIÓ DE PECES			
DESCRIPCIÓ	QUANTITAT	PREU	SUBTOTAL
TALL LÀSER ALUMINI	-	-	61,50 €
TALL FUSTA TALLER	3h	23€/h	69,00 €
TALL ALUMINI TALLER	3h	20€/h	60,00 €
SUBTOTAL FABRICACIÓ			190,50 €

PRESSUPOST TOTAL	
DESCRIPCIÓ	SUBTOTAL
ELECTRÒNICA	211,66 €
MECÀNICA	146,72 €
FABRICACIÓ	190,50 €
TOTAL FABRICACIÓ	
548,87 €	

El cost teòric de la meva feina com a enginyer es podria calcular com:

DESCRIPCIÓ	ASSIGNATURA	CÀRREGA	HORES / ECTS	TOTAL HORES	PREU	TOTAL ENGINYERIA
ENGINYERIA	TFG	24 ECTS	25	600	20€/h	12.000 €

CROQUIS FINALS ACOTATS AMB VARIABLES

Per poder vincular totes les cotes entre elles de manera que s'actualitzessin automàticament al modificar-ne alguna, es van dibuixar a mà totes les peces i se'ls hi van assignar cotes variables en forma de lletres.

Aquestes cotes variables s'afegien posteriorment en un full Excel on se'ls assignava un valor. Aquest valor podia provenir de tres maneres diferents:

1. D'una formula que s'escrivia seguint la geometria de les peces i el muntatge en el que es disposaven.
2. D'una cota mare, es a dir, una cota de la mateixa o d'una altre peça que definia directament la cota en qüestió.
3. D'una cota manual. En aquest cas la cota s'assignava manualment, directament amb un valor numèric. Aquest valor podia provenir de les mides dels objectes comprats o de decisions preses durant el disseny.

Aquestes cotes variables es van utilitzar també per acotar els dibuixos del 3D. De manera que al vincular el full d'Excel i el 3D, els valors de cada cota s'actualitzaven automàticament.

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

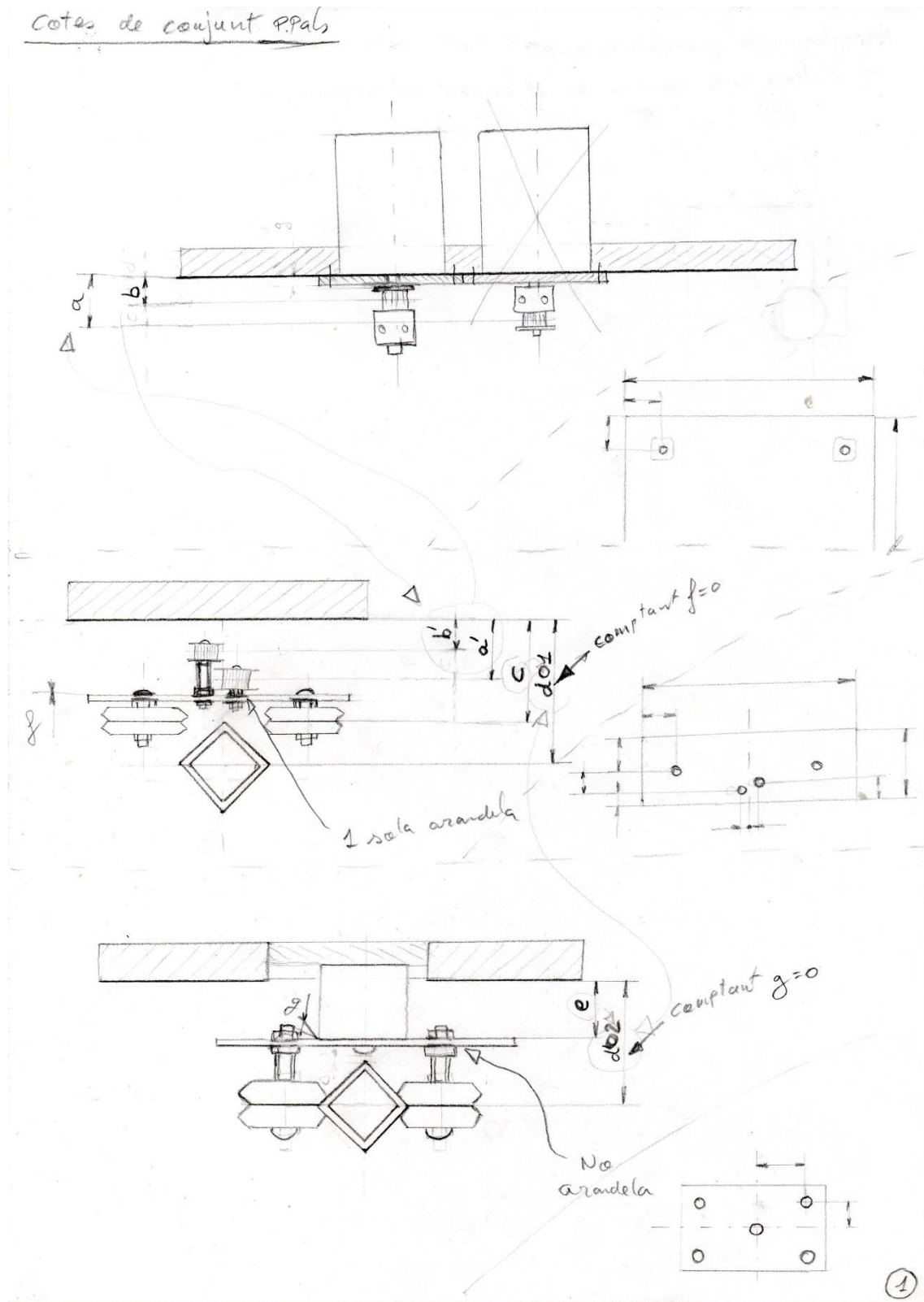


Figura 138. Esquemes manuals geomètrics principals. (Font: Elaboració pròpia)

Ajustar cada "geometria Principal" "c" i unicular - l_g
a l'altura de la ranura en el "suport de la guia fixe"

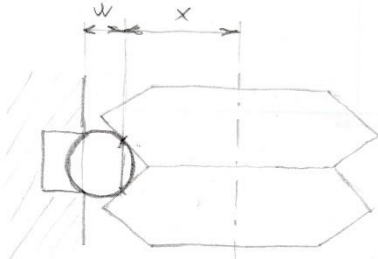


Figura 139. Esquema manual del contacte suport_guia_fixe - guia_fixe - roda. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

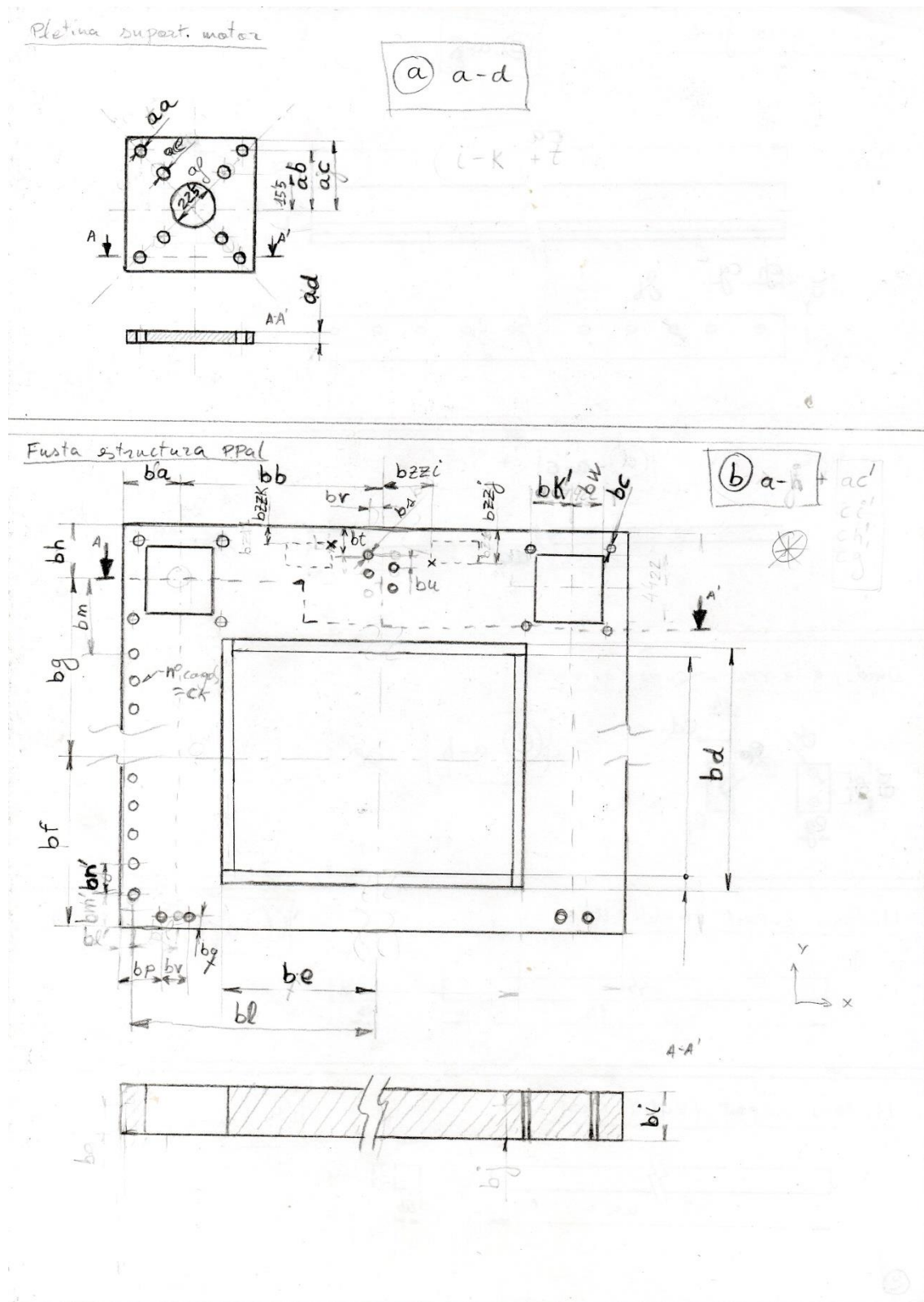


Figura 140. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

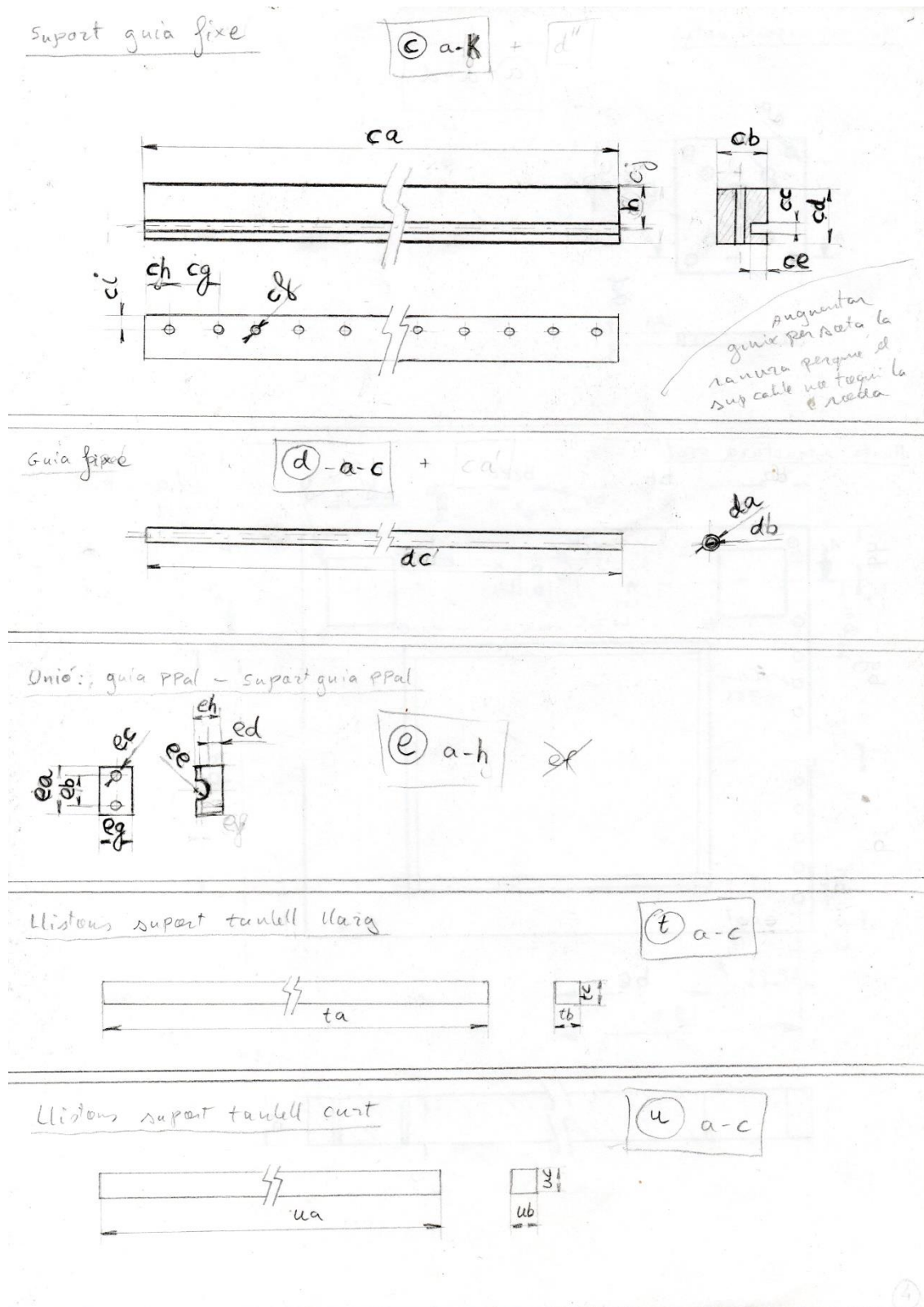


Figura 141. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

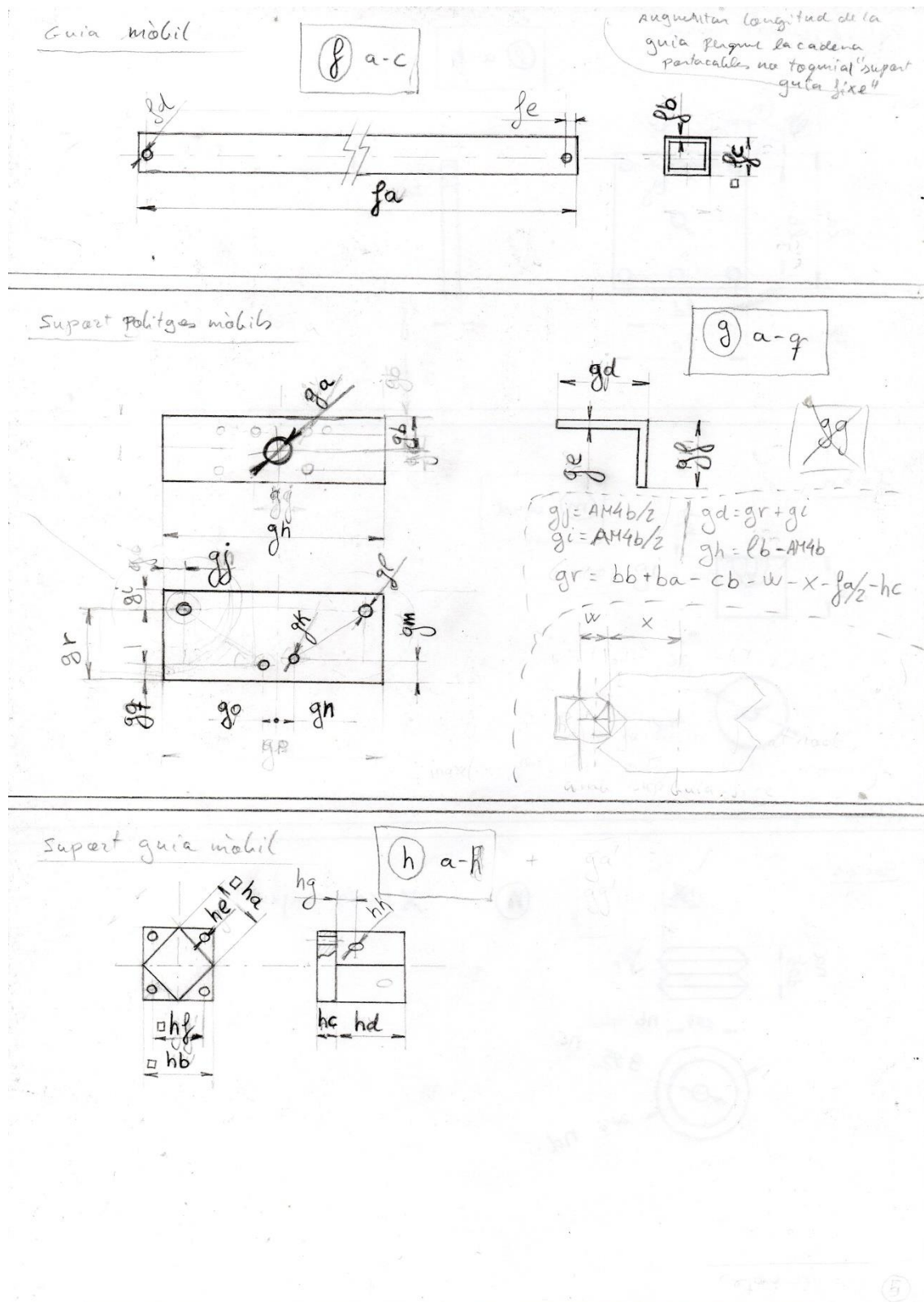


Figura 142. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

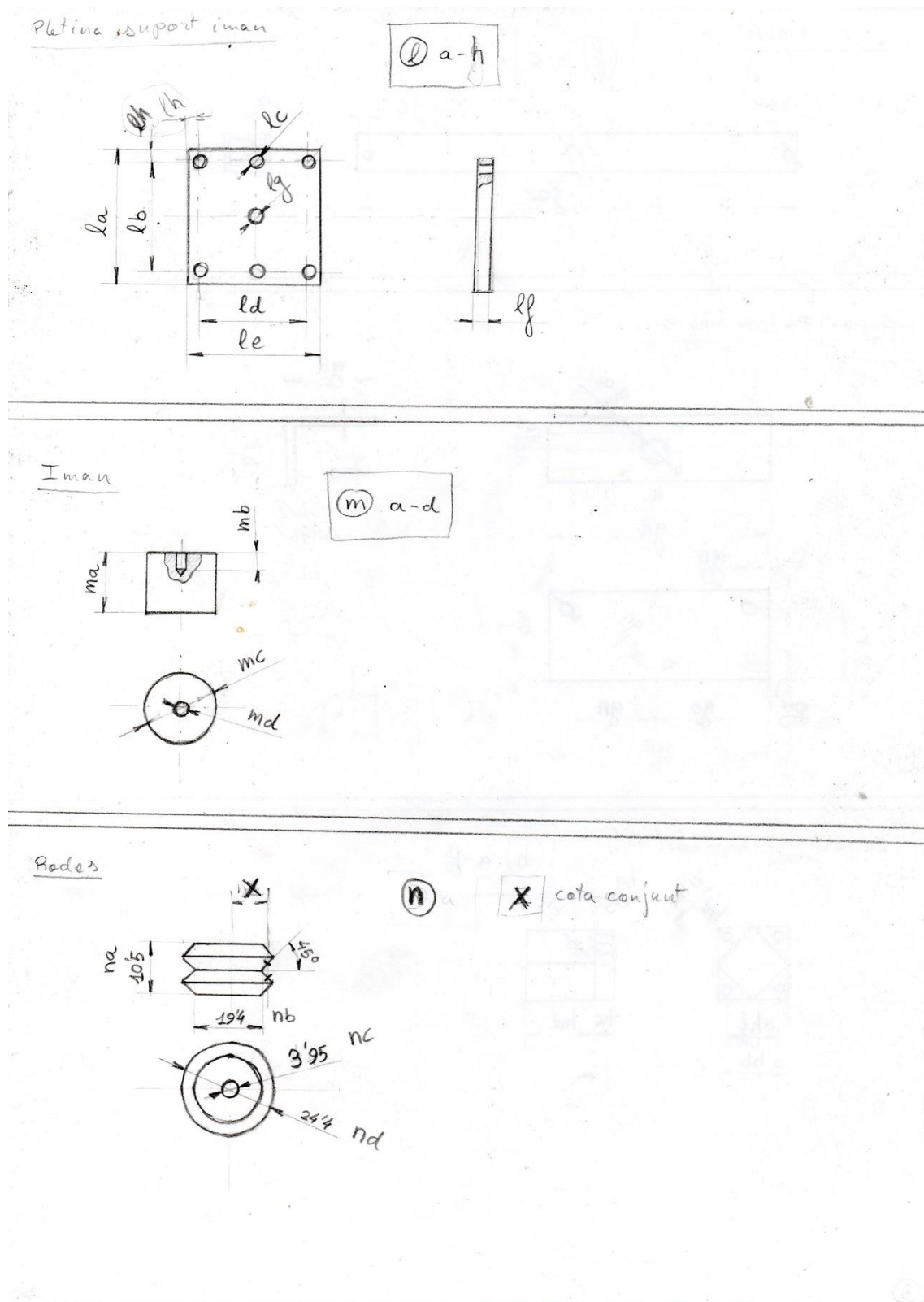


Figura 143. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

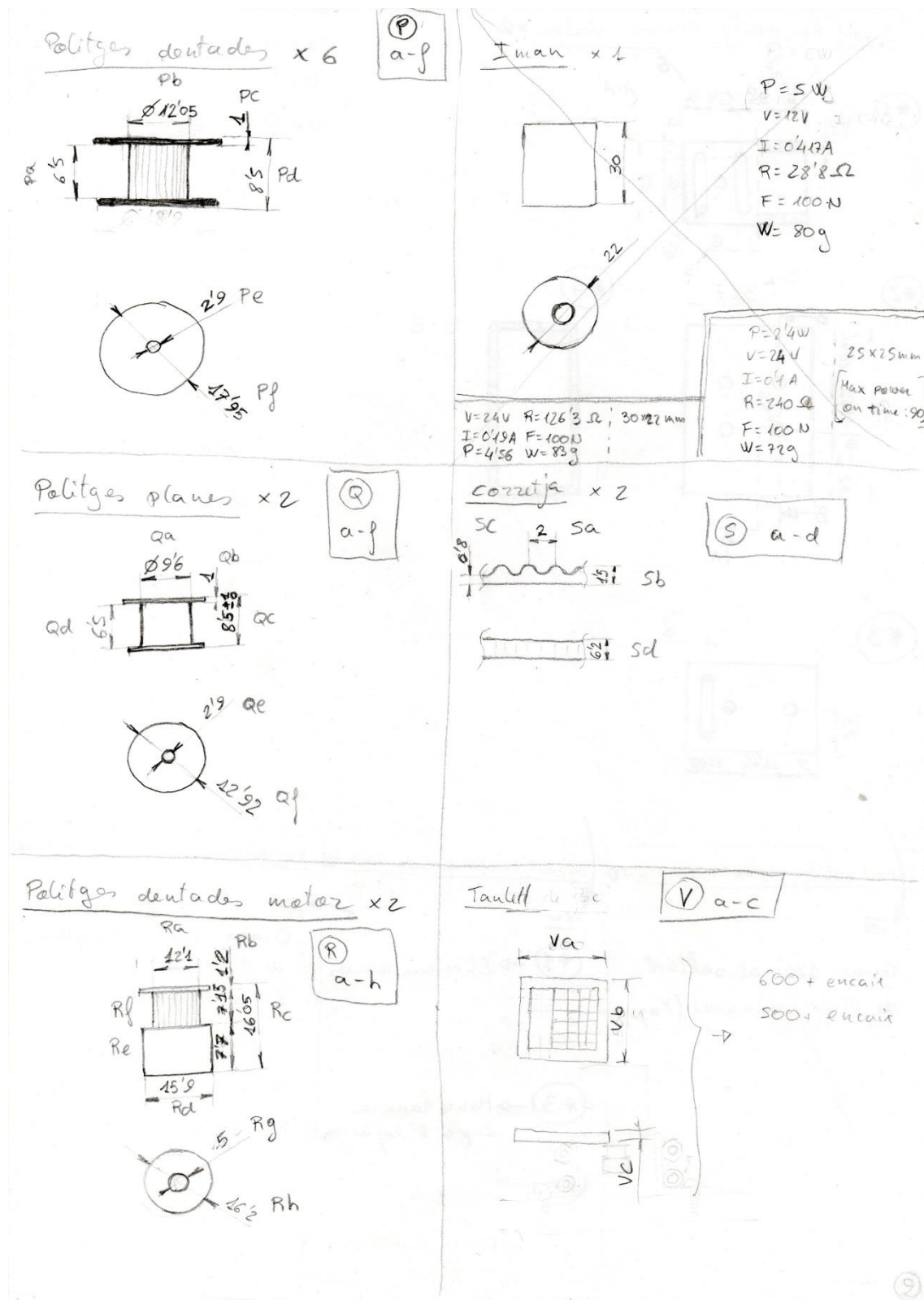


Figura 144. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

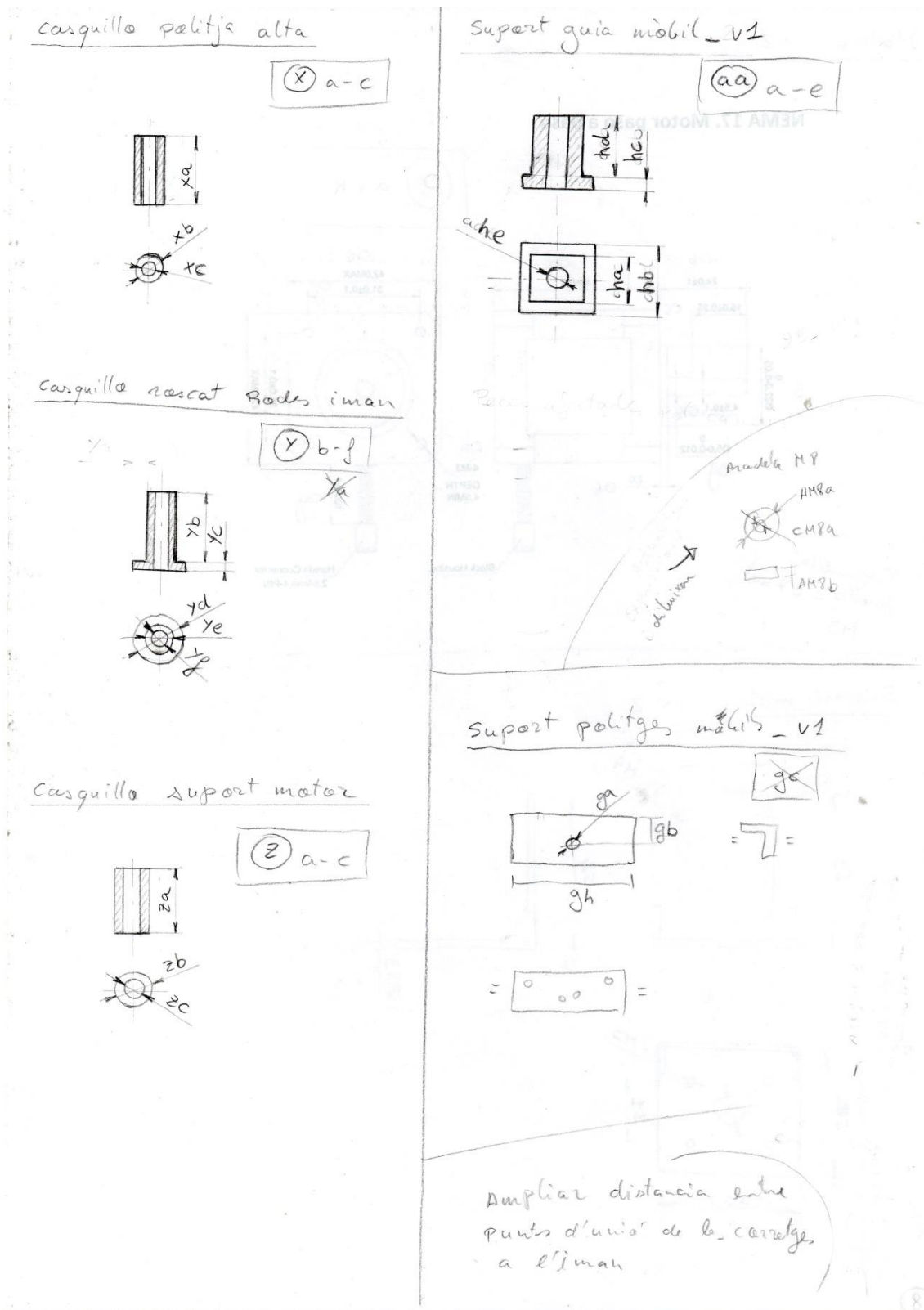


Figura 145. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

Mater $\times 2$

NEMA 17. Motor paso a paso

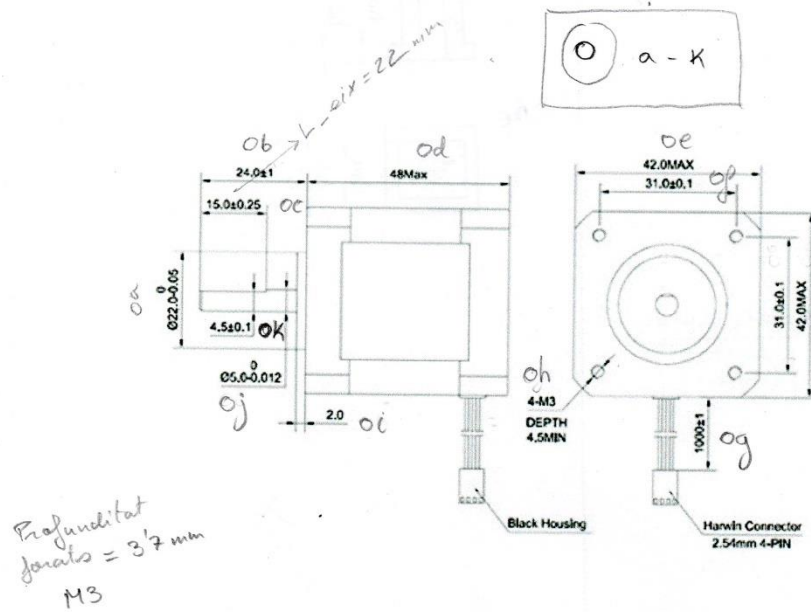


Figura 146. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

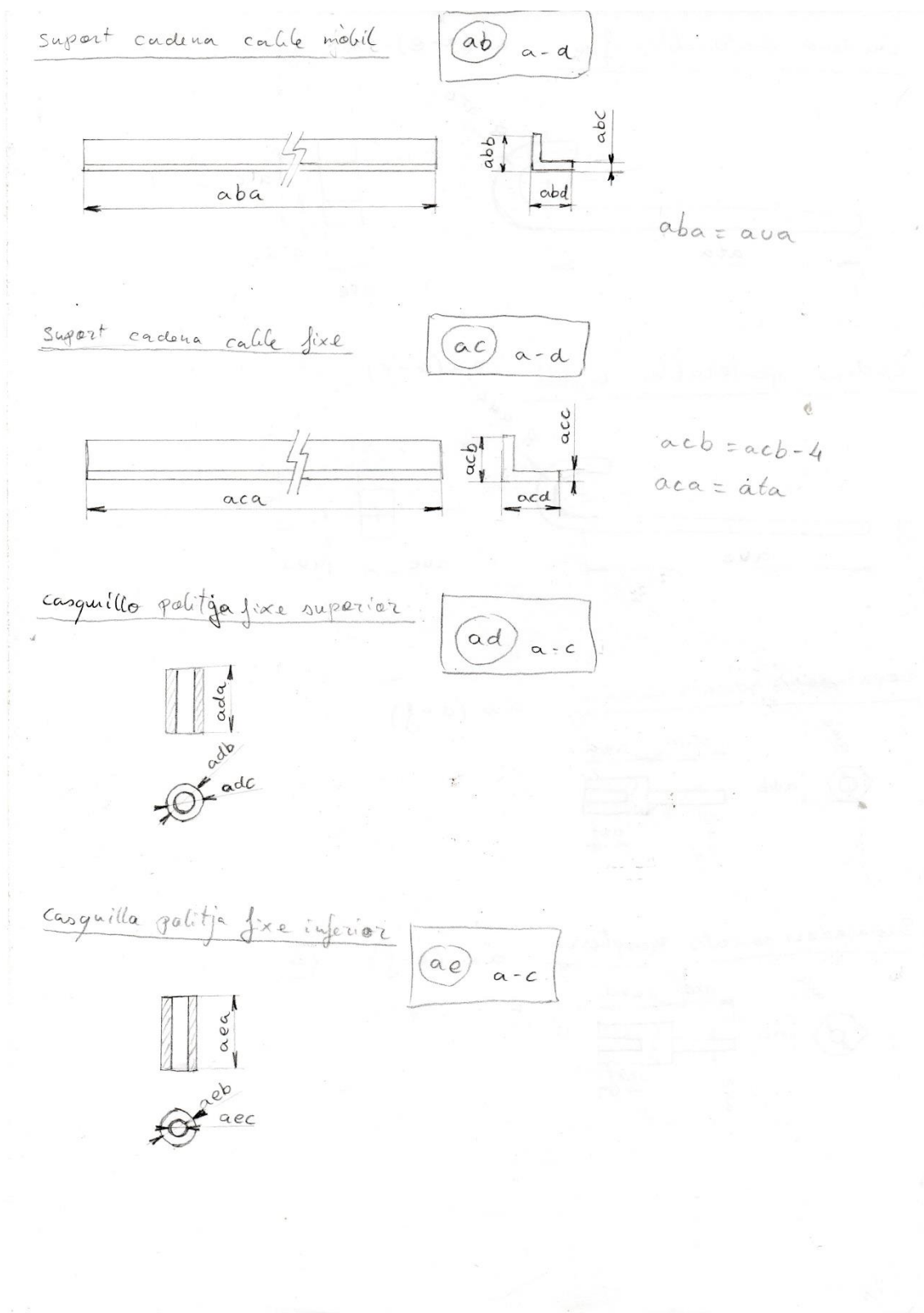
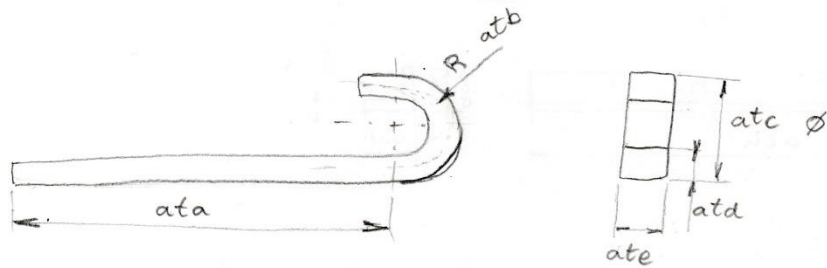
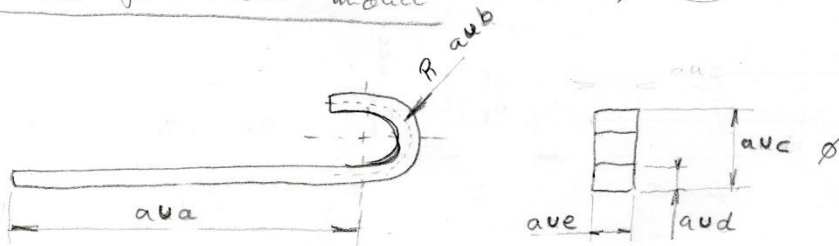


Figura 147. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

Cadena portacables fixe $at(a-e)$ (*)

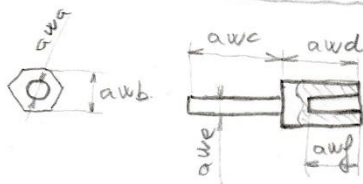


Cadena portacables mòbil $au(a-e)$ (*)



Separadors roscats càmera

$aw(a-f)$ (*)



Separadors roscats Raspberry

$ax(a-f)$ (*)

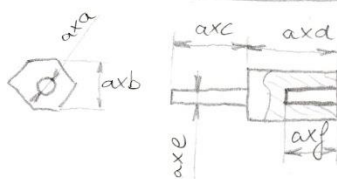


Figura 148. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

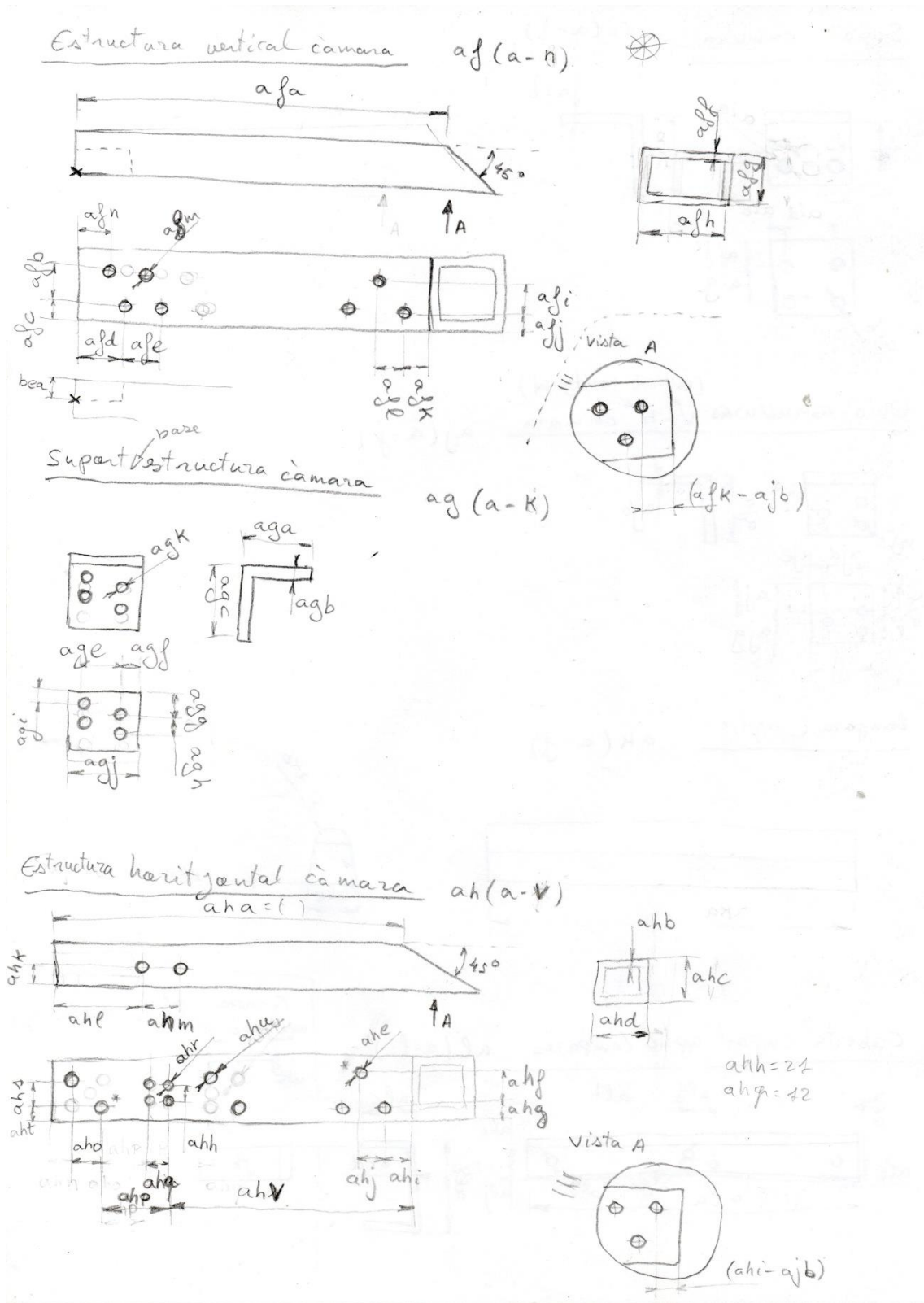


Figura 149.. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

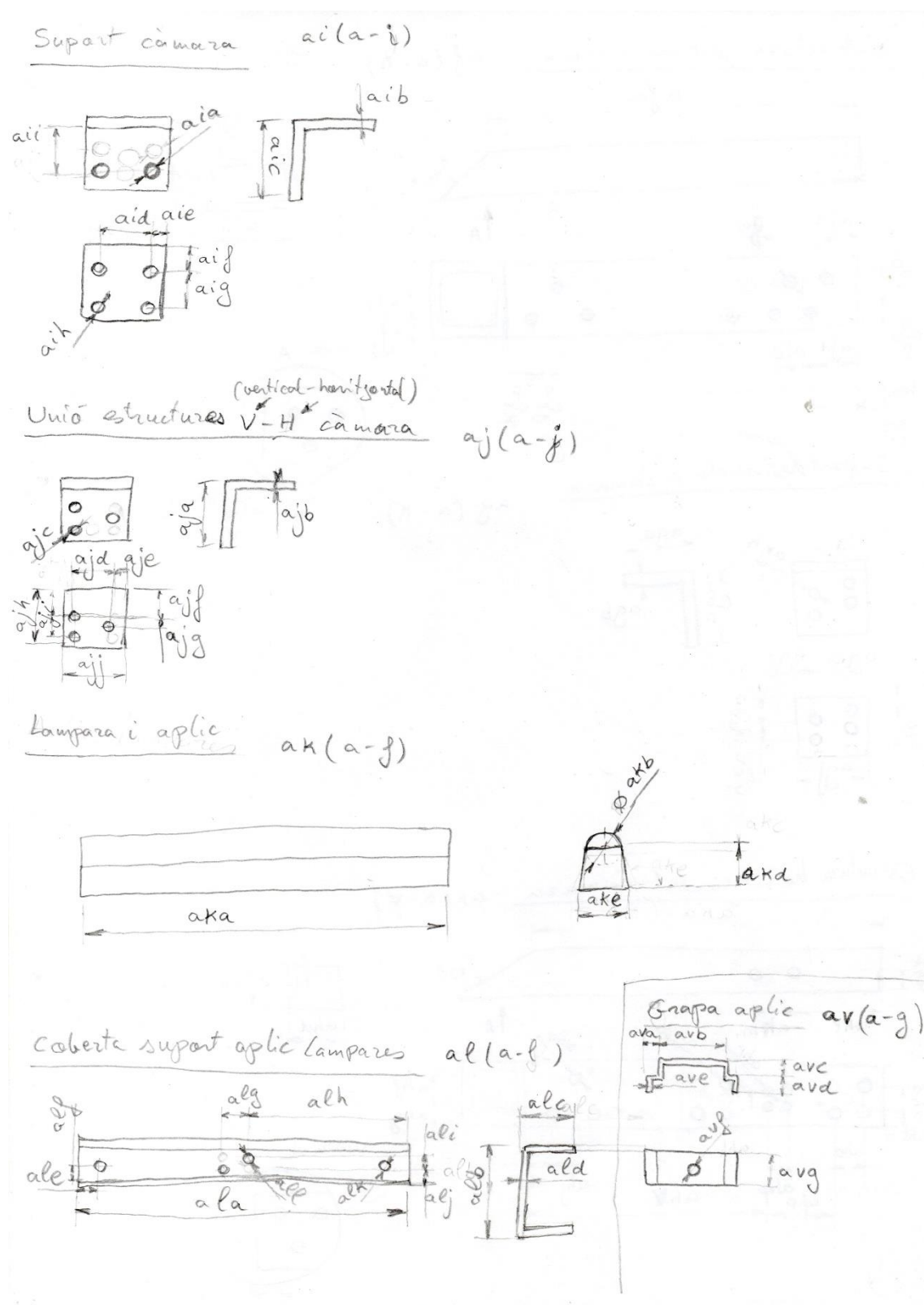


Figura 150. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

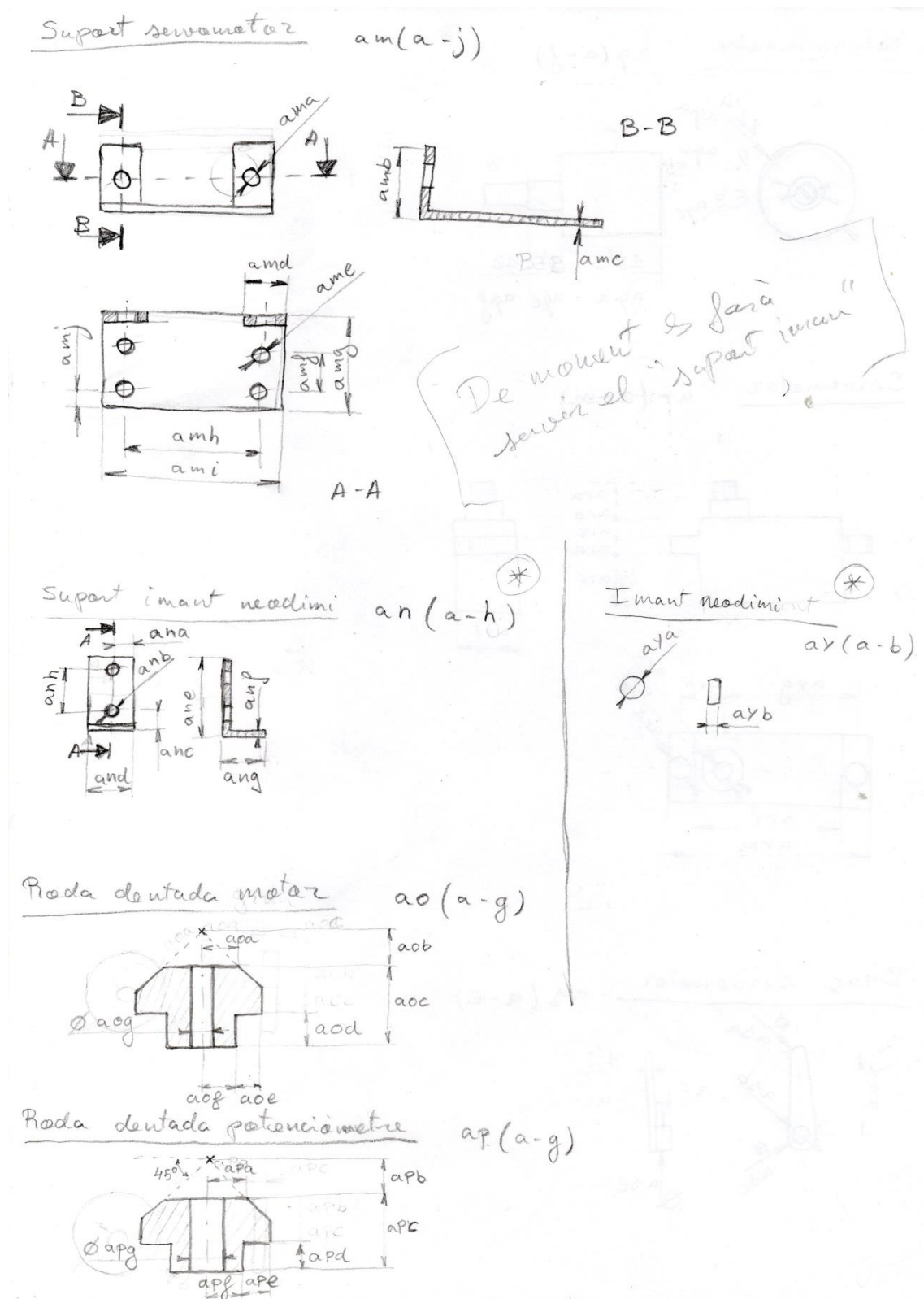
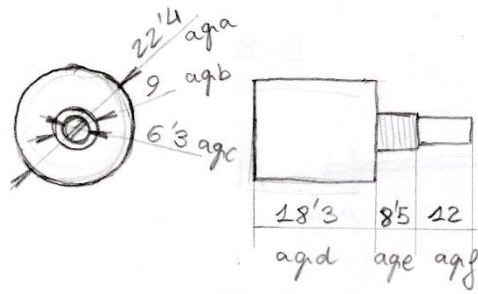


Figura 151. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

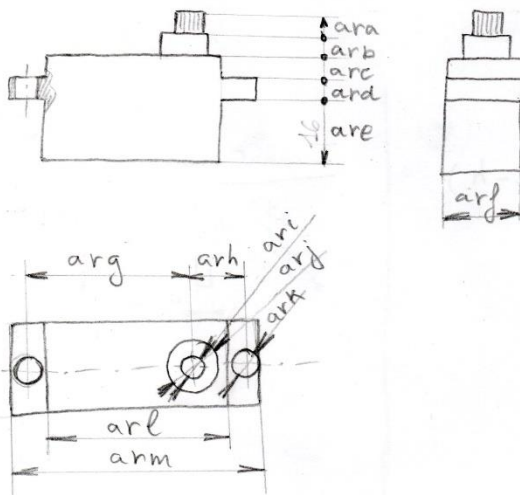
Potenciò metre

ag(a-f)



Servomotor

ar(a-m)



Brac servomotor

as(a-e)

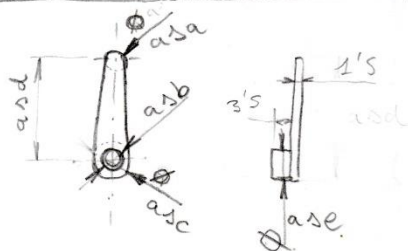


Figura 152. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

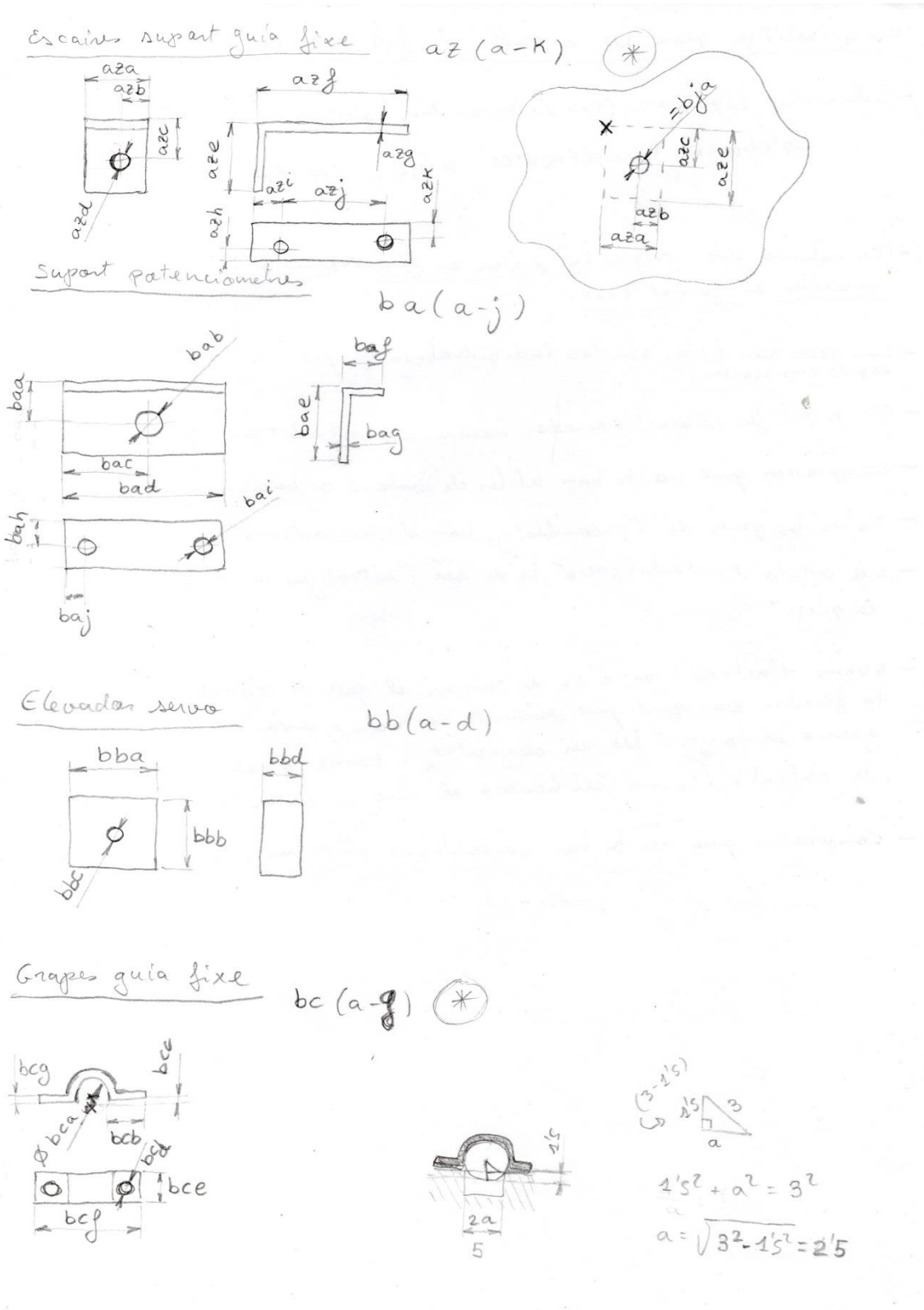


Figura 153. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

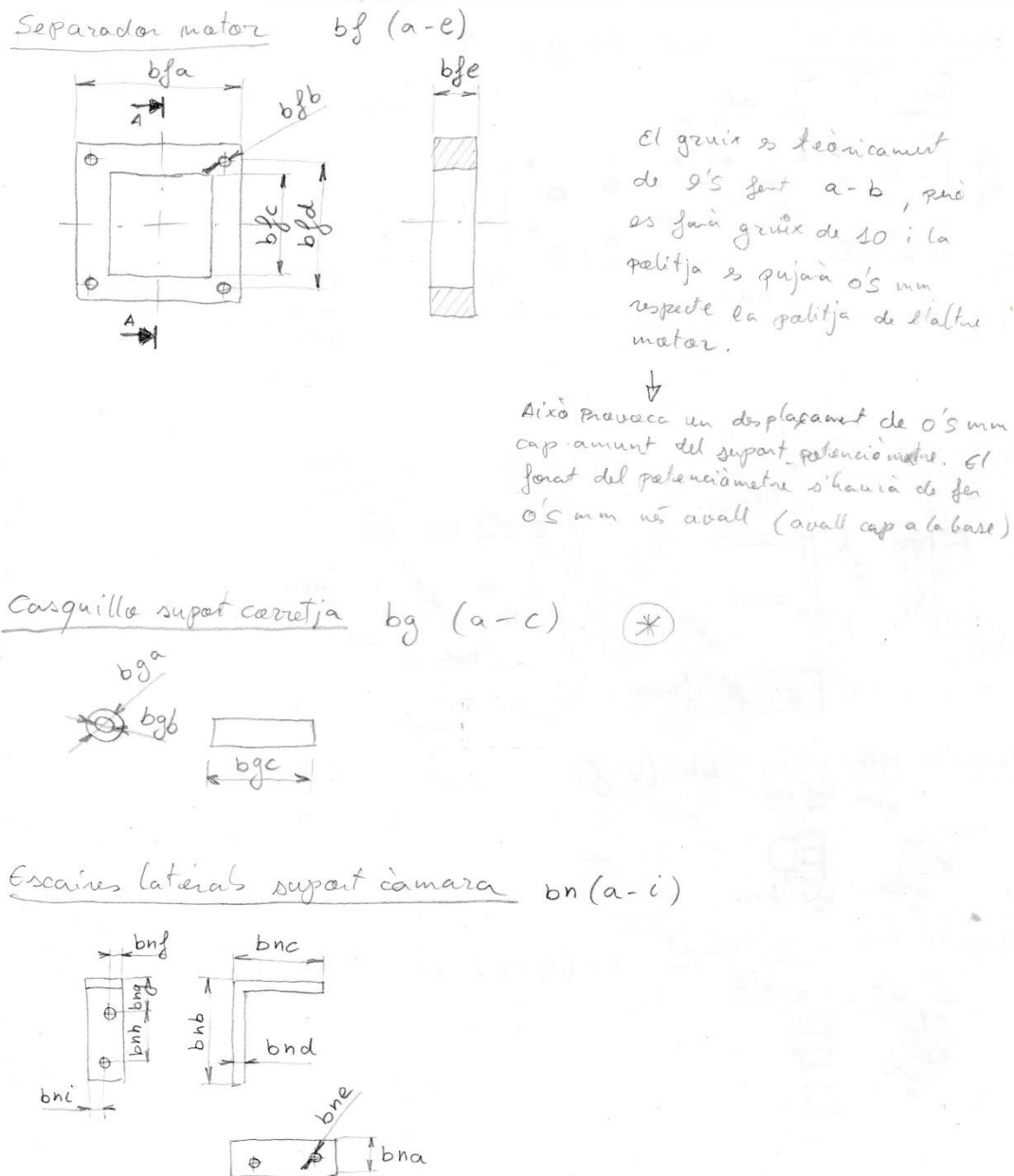


Figura 154. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

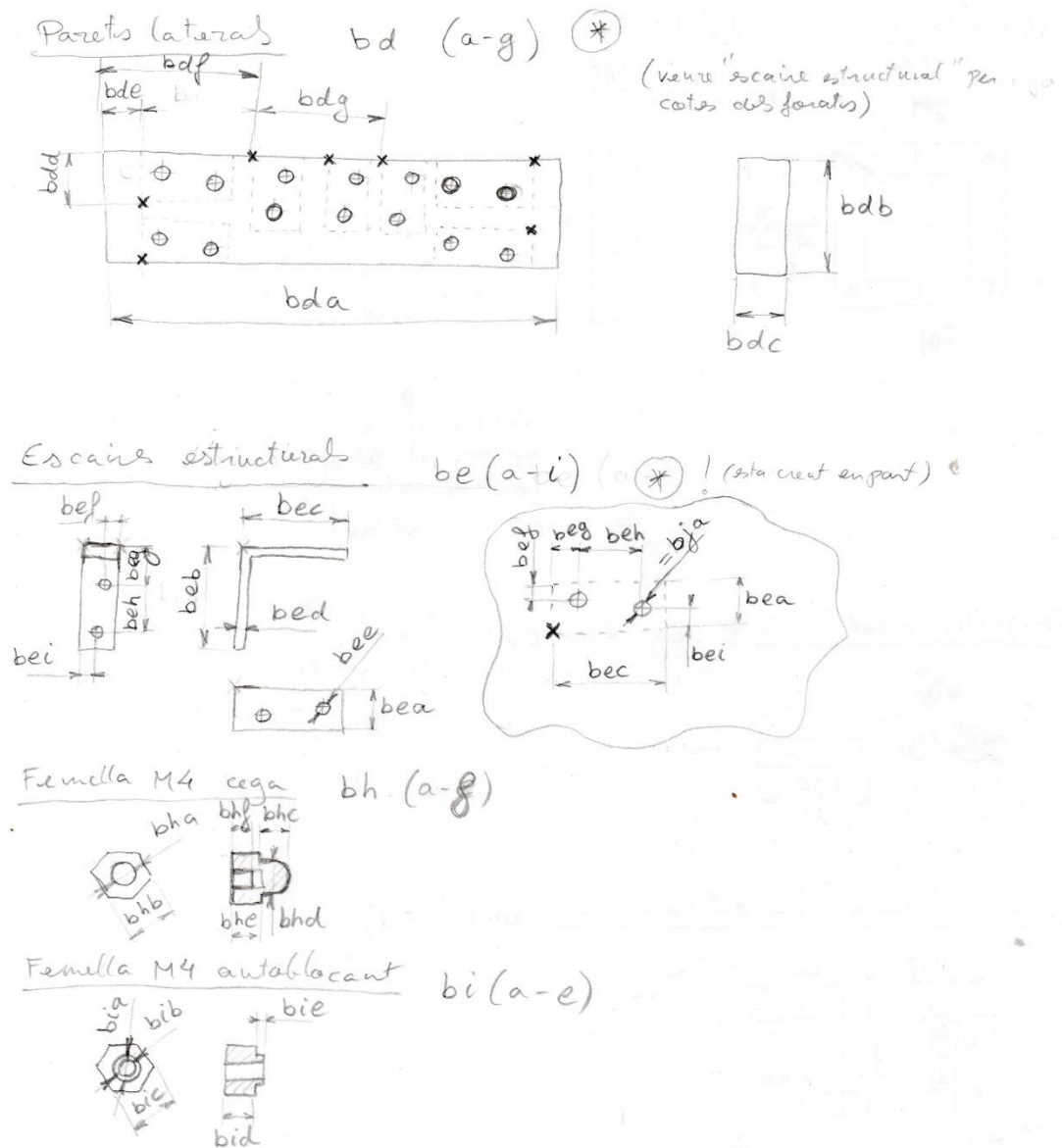


Figura 155. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

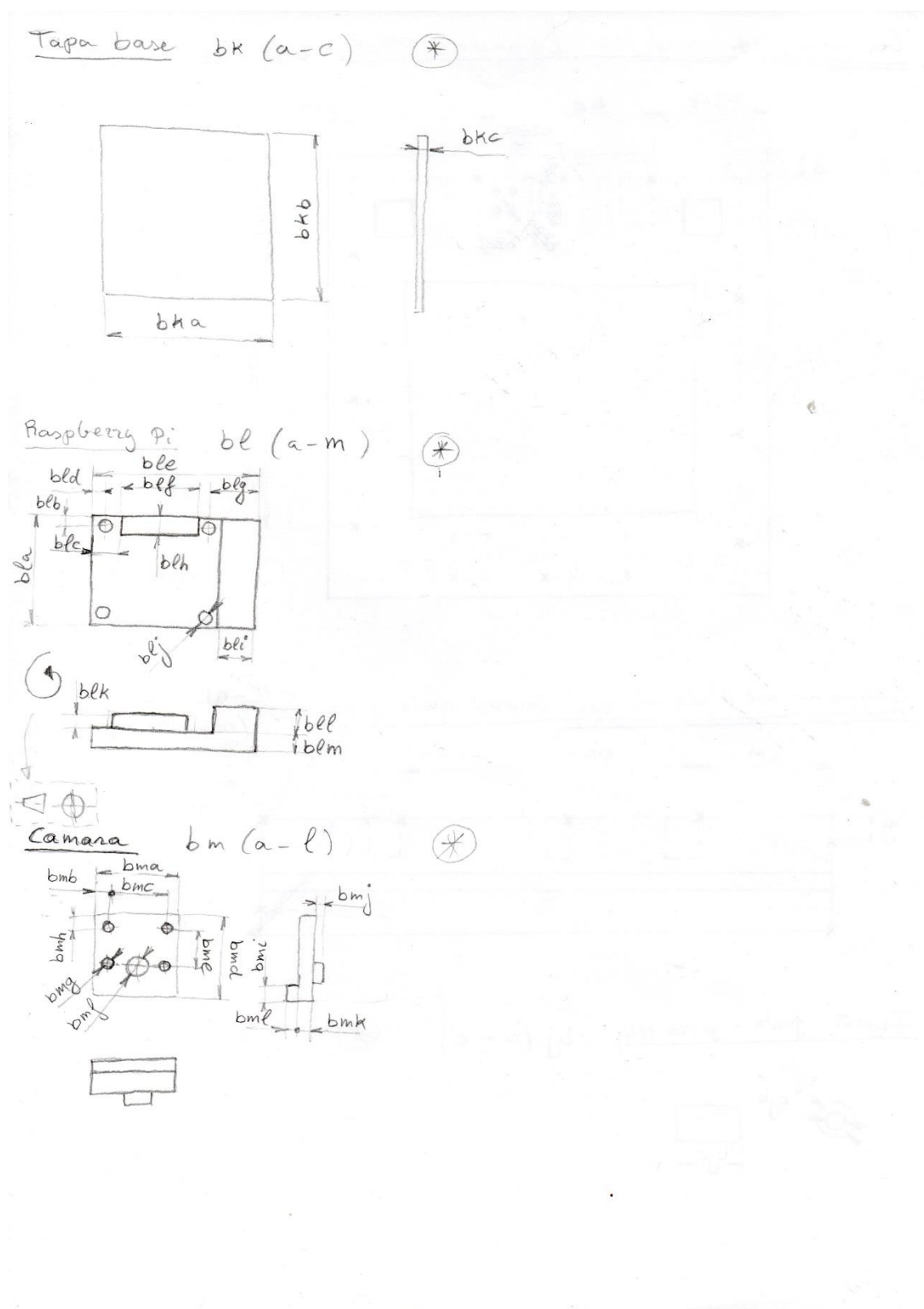


Figura 156. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

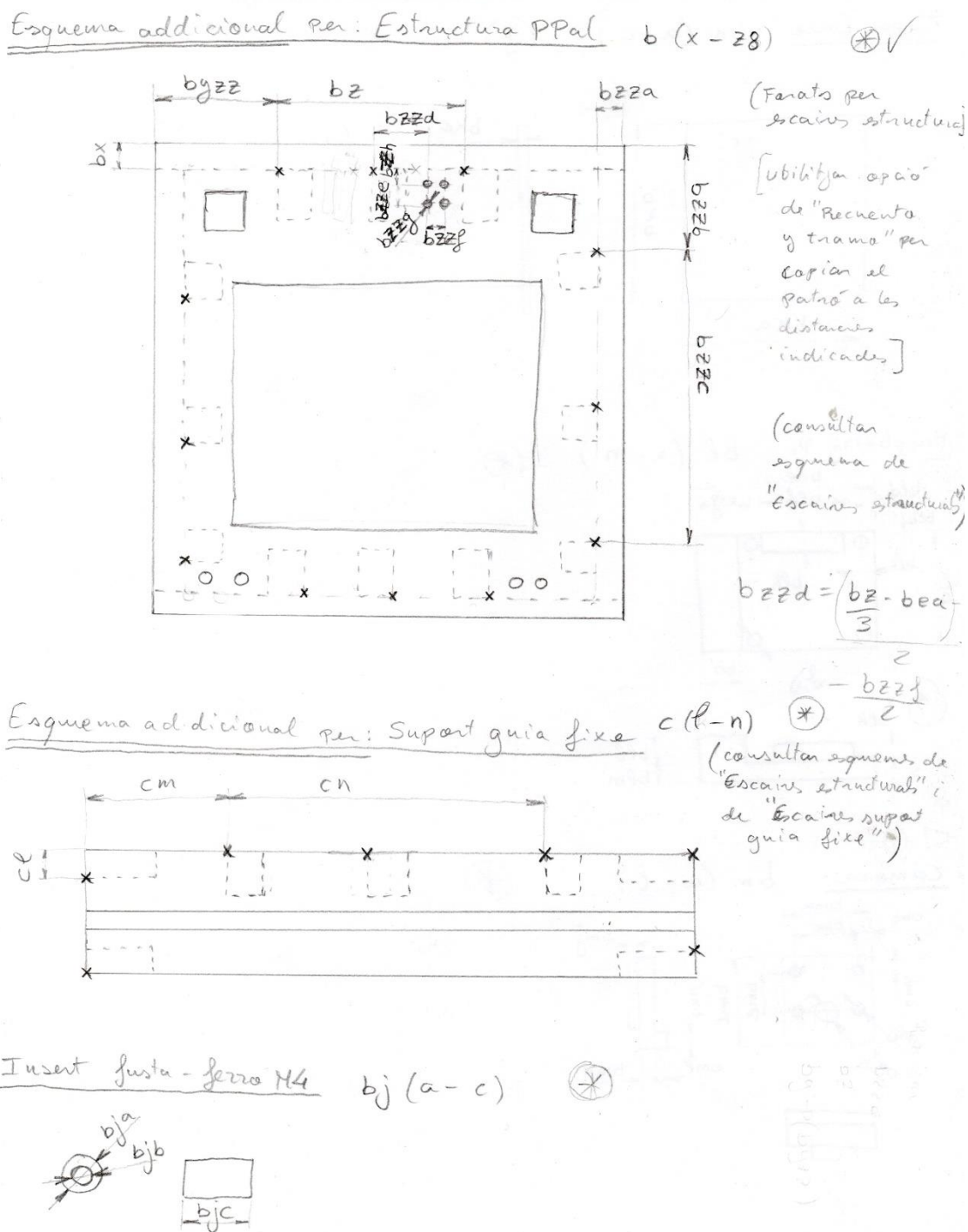


Figura 157. Esquemes manuals acotats amb variables. (Font: Elaboració pròpia)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

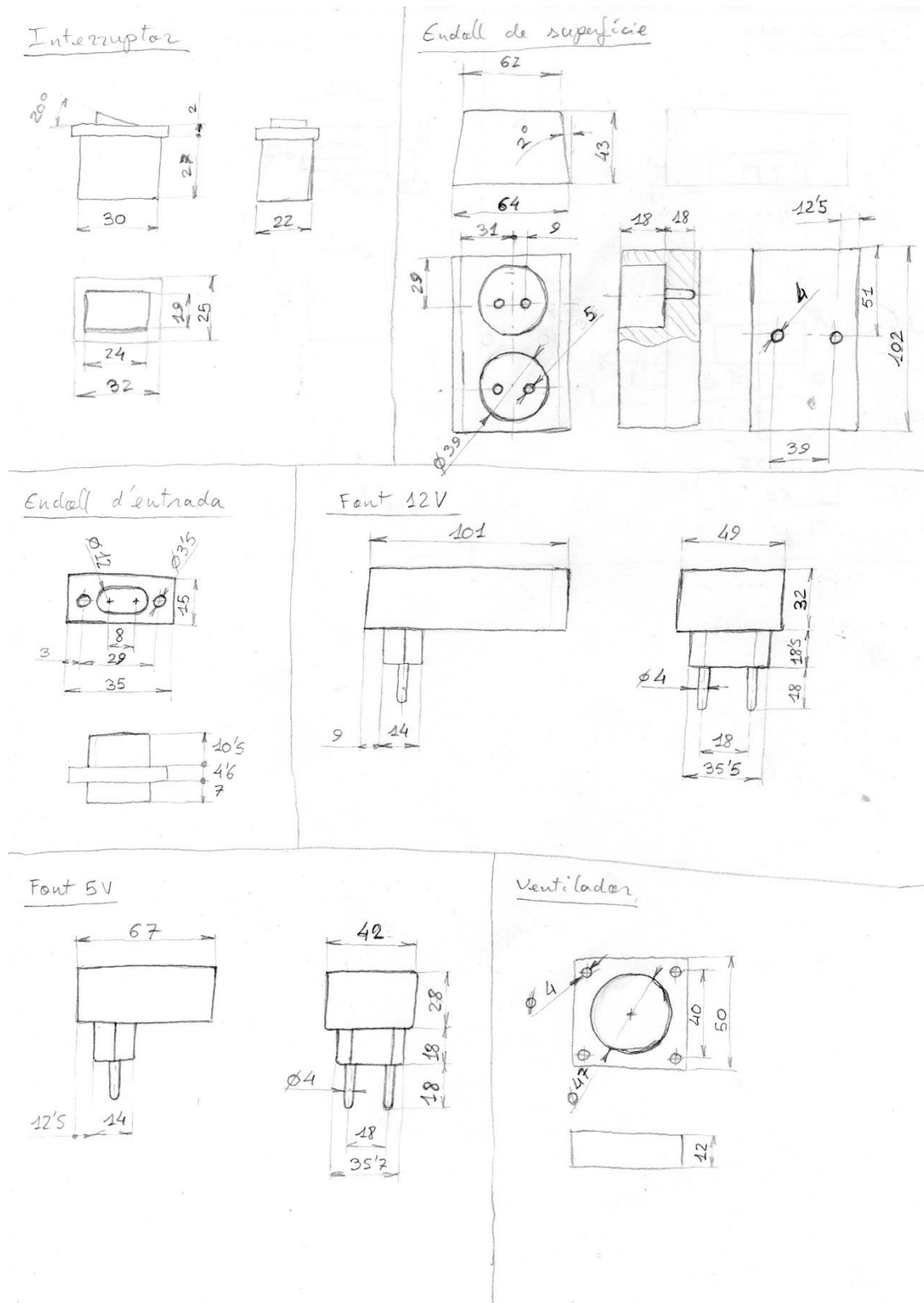


Figura 158. Esquemes manuals acotats amb valors. (Font: Elaboració pròpia)

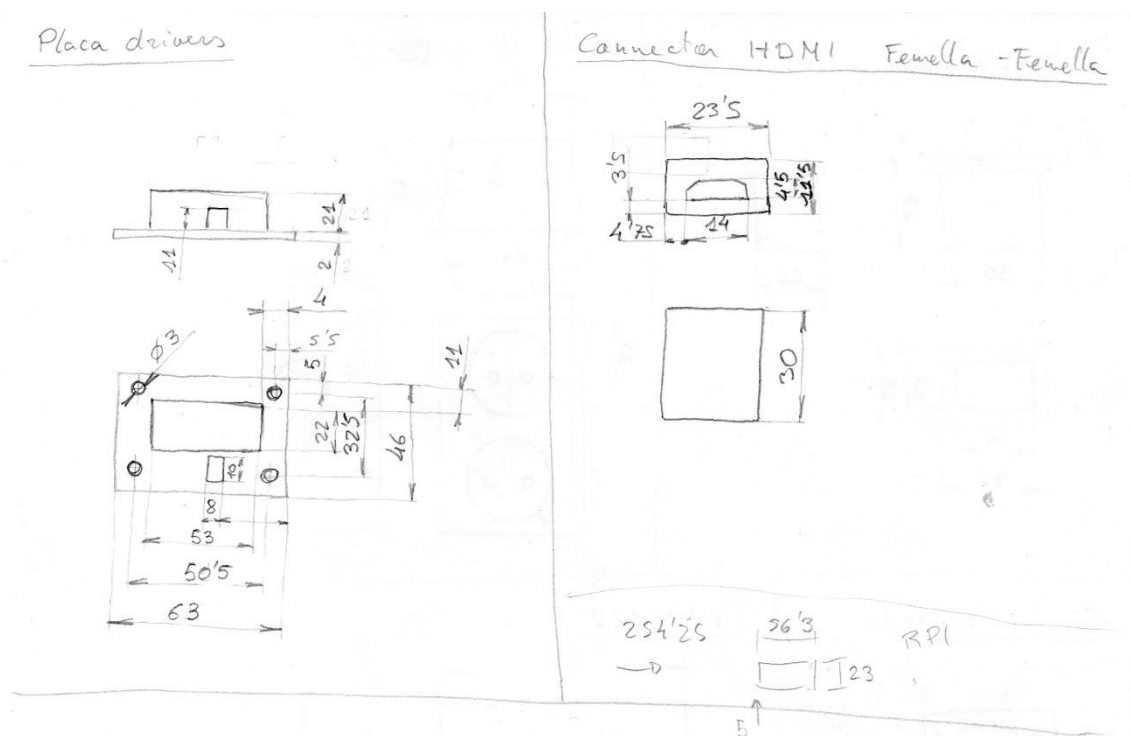


Figura 159. Esquemes manuals acotats amb valors. (Font: Elaboració pròpia)

FULL DE CàLCUL

A partir dels esquemes acotats amb variables es va crear un full de càlcul dins l'arxiu de conjunt 3D en NX.

Gràcies a utilitzar un full de càlcul amb les cotes relacionades amb fórmules, cada vegada que es modificava una cota, automàticament es re-calculaven totes les cotes de la màquina.

A partir d'aquí les cotes juntament amb els valors de cada cota s'introduïen com a expressions dins el conjunt 3D d'NX.

Un cop introduïdes les expressions es podien dibuixar les diferents peces en 3D fent servir les cotes anotades en els esquemes manuals.

La introducció i actualització de les cotes des del full de càlcul al 3D es lent però automàtic.

Un altre benefici que comporta fer servir cotes variables en un full de càlcul és poder acotar el 3D amb cotes que depenen d'altres peces encara que aquestes peces encara no estiguin dibuixades fent servir un valor temporal.

Dissenyar fent servir aquest sistema és força laboriós, però s'eviten molts errors de càlcul, i si es cometen, es poden rectificar ràpidament sense haver de re-calculat totes les cotes que depenen de l'anterior. A més es un sistema força flexible, de manera que es poden introduir canvis en les cotes i sense haver de re-calculat tot l'arbre de peces.

Malgrat tot, aquest sistema també suposa un inconvenient, si es canvia el concepte de muntatge o de funcionament, de manera que peces senceres siguin substituïdes per peces completament diferents amb geometries poc semblants, és possible que s'hagin de reescriure les formules geomètriques d'altres peces vinculades.

Fent servir aquest sistema basat en la flexibilitat i en l'automatització dels càlculs, pretenia poder actualitzar contínuament el disseny i així crear un prototip inicial molt pròxim al prototip final, de manera que no hagués de fabricar-ne diversos amb el cost econòmic que això suposa.

Seguidament adjunto la imatge del full de càlcul. Com que esta enganxat en un document de text no es poden veure les formules, però es pot veure a l'interior del conjunt 3D (que adjuntaré al treball) en l'apartat full de càlcul.

NOM PEÇA	ID COTA	COTA MARE (ID)	FÓRMULA	VAL. MANUAL	VAL. COTA
Cargol_M3	CM3a			3	3.00
	CM3b			6	6.00
	CM3c		20.00		20.00
	CM3e		30.00		30.00
	CM3f			50	50.00
	CM3d			3	3.00
Femella_M3	FM3a			2.3	2.30
	FM3b			5.5	5.50
Arandela_M3	AM3a			0.8	0.80
	AM3b			7	7.00
Cargol_M4	CM4a			4	4.00

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

	CM4b			8	8.00
	CM4f				0.00
	CM4g				0.00
	CM4d		20.00		20.00
	CM4e		30.00		30.00
Femella_M4	FM4a			2.9	2.90
	FM4b			7	7.00
Arandela_M4	AM4a			1	1.00
	AM4b			11.5	11.50
Cargol_M6	CM6a			6	6.00
	CM6b			10	10.00
	CM6c			4	4.00
	CM6d			35	35.00
Femella_M6	FM6a			5	5.00
	FM6b			10	10.00
Arandela_M6	AM6a			12	12.00
	AM6b			1.2	1.20
Cargol_M8	CM8a			8	8.00
	CM8b			13.8	13.80
	CM8c			5.5	5.50
	CM8d			40	40.00
Femella_M8	FM8a			6.25	6.25
	FM8b			12.8	12.80
Arandela_M8	AM8a			15.8	15.80
	AM8b			1.5	1.50
Cargols fusta 3mm	CF3a			3.1	3.10
	CF3b			2.2	2.20
	CF3c			5.7	5.70
	CF3d				0.00
	CF3e				0.00
	CF3f				0.00
Politja_dentada	pa			6.5	6.50
	pb			12	12.00
	pc			1	1.00
	pd			8.5	8.50
	pe	CM3a			3.00
	pf			18	18.00
Politja_plana	qa			12	12.00
	qb			1	1.00
	qc			8.5	8.50
	qd			6.5	6.50
	qe	CM3a			3.00
	qf			18	18.00
Politja_motor	ra			37.9	37.90
	rb			1.5	1.50

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

	rc			16	16.00
	rd			25	25.00
	re			6	6.00
	rf			7	7.00
	rg			5	5.00
	rh			43.9	43.90
Corretja	sa			2	2.00
	sb			1.5	1.50
	sc			0.8	0.80
	sd			6.2	6.20
Motor	oa			22	22.00
	ob			24	24.00
	oc			18	18.00
	od			37.3	37.30
	oe			42	42.00
	of			31	31.00
	og			1000	1000.00
	oh	CM3a			3.00
	oi			2	2.00
	oj			5	5.00
	ok			4.5	4.50
Roda	na			10.5	10.50
	nb			19.4	19.40
	nc	CM4a			4.00
	nd			24.4	24.40
Iman	ma			25.15	25.15
	mb			25.15	25.15
	mc	arl			22.30
	md		4.00		4.00
Geometria_general (ZC)	a		20.50		20.50
	b		11.00		11.00
	f		0.00		0.00
	c		40.00		40.00
	dpoli		43.07		43.07
	dima		35.10		35.10
	d		43.07		43.07
	e		15.15		15.15
	h		43.40		43.40
	g		7.97		7.97
	v			1	1.00
	w		3.78		3.78
	x		11.82		11.82
	y			540	540.00
	z			450	450.00
Suport_iman (Retall de planxa d'acer)		la		47.30	47.30

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

	lb		35.80		35.80
	lc	nc			4.00
	ld		47.68		47.68
	le		59.18		59.18
	lf			1.5	1.50
	lg	md			4.00
	lh		5.75		5.75
Casquillo_roscat	yb			12.2	12.20
	yc			1	1.00
	yd			9.1	9.10
	ye			6	6.00
	yf	CM4a			4.00
Imant_neodimi	aya			6	6.00
	ayb			3	3.00
Servomotor	ara			4	4.00
	arb			4.5	4.50
	arc			4	4.00
	ard			2.5	2.50
	are			15.7	15.70
	arf			12.4	12.40
	arg			18	18.00
	arh			9	9.00
	ari			4.8	4.80
	arj			11.5	11.50
	ark			2	2.00
	arl			22.3	22.30
	arm			32.3	32.30
Braç_servo	asa			4.8	4.80
	asb	ari			4.80
	asc			7	7.00
	asd			14	14.00
	ase			6.8	6.80
Cadena_pcab_mobil	aua			320	320.00
	aub			10	10.00
	auc			28	28.00
	aud			8	8.00
	aue			8.7	8.70
Guia_mobil (Tub quadrat d'alumini)	fa		643.65		643.65
	fb			1.3	1.30
	fc			20	20.00
	fd	CM3a			3.00
	fe		9.75		9.75
Suport_guia_mòbil_1	ha		17.40		17.40
	hb			20.4	20.40
	hc			5	5.00

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

	hd			19.5	19.50
	he	CM8a			8.00
Suport_politges_mobils_dreta_v1 (Perfil "L" d'acer)	ga	CM8a			8.00
	gb		14.42		14.42
	gd		71.85		71.85
	ge			3	3.00
	gf		28.85		28.85
	gh		59.18		59.18
	gi		5.75		5.75
	gj		5.75		5.75
	gk	CM3a			3.00
	gl	CM4a			4.00
	gm		10.00		10.00
	gn		6.40		6.40
	go		6.40		6.40
	gr		66.10		66.10
	gq		10.00		10.00
Casquillo_pol_alta (Tub de plastic tallat)	xa		11.90		11.90
	xb			4.5	4.50
	xc	CM3a			3.00
Suport_cable_mobil	aba	fa			643.65
	abb			10	10.00
	abc			1.5	1.50
	abd			15	15.00
Cadena_pcab_fixe	ata			320	320.00
	atb			10	10.00
	atc			28	28.00
	atd			8	8.00
	ate			8.7	8.70
Suport_taulell_llarg (Llistó de fusta tallat)	ta		546.00		546.00
	tb			5	5.00
	tc		10.00		10.00
Suport_taulell_curt (Llistó de fusta tallat)	ua		456.00		456.00
	ub	tb			5.00
	uc	tc			10.00
Taulell ()	va		550.00		550.00
	vb		460.00		460.00
	vc			5	5.00
Suport_motor (Retall de planxa d'acer)	aa	CM4a			4.00
	ab		31.20		31.20
	ac		36.95		36.95

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

	ad			3	3.00
	ae	oh			3.00
	af		22.10		22.10
Subjeccio_guia_fixe (Impresió 3D)	ea		22.00		22.00
	eb		15.00		15.00
	ec	CF3a			3.10
	ed			4	4.00
	ee		3.00		3.00
	eg		8.00		8.00
	eh		7.00		7.00
Guia_fixe (Tub rodó d'alumini)	da			6	6.00
	db			1	1.00
	dc		548.08		548.08
Suport_guia_fixe (Retall del tauler de contraxapat utilitzat en l'estructura ppal)	ca		665.53		665.53
	cb	bi			15.00
	cc			5	5.00
	cd			65	65.00
	ce			2	2.00
	cf	CF3a			3.10
	cg		66.55		66.55
	ch		33.28		33.28
	ci		6.50		6.50
	cj	h			43.40
	ck			10	10.00
	cl	bea			19.30
	cm		92.90		92.90
	cn	bzzc			460.43
Estructura_ppal (Tauler de contraxapat)	ba		60.95		60.95
	bb		362.58		362.58
	bc	aa			4.00
	bd		461.00		461.00
	be		275.50		275.50
	bf		302.54		302.54
	bg		332.04		332.04
	bh	ba			60.95
	bi			15	15.00
	bj			9	9.00
	bk	ab			31.20
	bl		417.03		417.03
	bm		71.23		71.23

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

	bp		48.00		48.00
	bq		32.00		32.00
	bn	cg			66.55
	bo			9	9.00
	br		10.00		10.00
	bs			5	5.00
	bt		48.00		48.00
	bv		38.70		38.70
	bw		25.45		25.45
	bu		10.00		10.00
	bx	bdc			15.00
	byzz		107.90		107.90
	bz		611.95		611.95
	bzza	cb			15.00
	bzzb		107.90		107.90
	bzzc		460.43		460.43
	bzzd		25.95		25.95
	bzze		58.00		58.00
	bzzf		49.00		49.00
	bzzg			2	2.00
	bzzh	blg			25.50
	bzzi		20.00		20.00
	bzzj		35.30		35.30
	bzzk		16.00		16.00
Casquillo_suport_motor	za		1.70		1.70
	zb			6	6.00
	zc			4	4.00
Suport_cable_fix	aca		274.04		274.04
	acb			20	20.00
	acc			1.5	1.50
	acd			40	40.00
Casquillo_pol_fix_sup (Tub de plastic tallat)	ada		3.65		3.65
	adb			4.5	4.50
	adc	CM3a			3.00
Casquillo_pol_fix_inf (Tub de plastic tallat)	aea		13.15		13.15
	aeb			4.5	4.50
	aec	CM3a			3.00
Estr_vert_cam	afa			450	450.00
	afb	age			20.00
	afc	agf			10.00
	afd	agg			22.00
	afe	agh			20.00
	aff			1.5	1.50
	afg			20	20.00

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

	afh			40	40.00
	afi	aji			20.00
	afj	aje			10.00
	afk	ajf			12.00
	afl	ajg			10.00
	afm	ajc			5.00
	afn	agi			12.00
Sup_estr_vert_cam	aga			50	50.00
	agb			2	2.00
	agc			50	50.00
	age			20	20.00
	agf			10	10.00
	agg			22	22.00
	agh			20	20.00
	agi			12	12.00
	agj			40	40.00
	agk			5	5.00
Estructura_hori_cam	aha			450	450.00
	ahb			1.5	1.50
	ahc			20	20.00
	ahd			40	40.00
	ahe	ajc			5.00
	ahf	ajd			20.00
	ahg	aje			10.00
	ahh	bmc			21.00
	ahí	ajf			12.00
	ahj	ajg			10.00
	ahk			10	10.00
	ahl			10	10.00
	ahm			15	15.00
	ahn			10	10.00
	aho	ali			10.00
	ahp		68.01		68.01
	ahq	bme			12.00
	ahr			1.5	1.50
	ahs			20	20.00
	aht			10	10.00
	ahu			5	5.00
	ahv		356.99		356.99
Unio_V-H_estr_cam	aja			40	40.00
	ajb			2	2.00
	ajc			5	5.00
	ajd			20	20.00
	aje			10	10.00
	ajf			12	12.00
	ajg			10	10.00

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

	ajh			40	40.00
	aji			20	20.00
	ajj			40	40.00
Lampara_i_aplic	aka			300	300.00
	akb			20	20.00
	akd			25	25.00
	ake			23	23.00
Coberta_lampara	ala			400	400.00
	alb			40	40.00
	alc			20	20.00
	ald			1	1.00
	ale			70	70.00
	alf		20.00		20.00
	alg	ahs			20.00
	alh		190.00		190.00
	ali		10.00		10.00
	alj			15	15.00
	alk	avf			4.00
	all	ahu			5.00
Roda_dent_motor	aoa		5.17		5.17
	aob	aoa			5.17
	aoc			10.83	10.83
	aod			5	5.00
	aoe		2.57		2.57
	aof			6	6.00
	aog	oj			5.00
Roda_dent_pot	apa		5.17		5.17
	apb	apa			5.17
	apc			10.83	10.83
	apd			5	5.00
	ape		2.57		2.57
	apf			6	6.00
	apg	aqc			6.30
Potenciometre	aqa			22.4	22.40
	aqb			9	9.00
	aqc			6.3	6.30
	aqd			18.3	18.30
	aqe			8.5	8.50
	aqf			12	12.00
Grapa aplic lampara	ava			6	6.00
	avb			10	10.00
	avc			3	3.00
	avd			4	4.00
	ave			0.5	0.50
	avf			4	4.00
	avg			9.5	9.50

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

Sup_rosc_cam	awa	CM3a			3.00
	awb			5	5.00
	awc			5	5.00
	awd			20	20.00
	awe	CM3a			3.00
	awf			7.5	7.50
Sup_rosc_Rasp	axa	CM3a			3.00
	axb			5	5.00
	axc			5	5.00
	axd			20	20.00
	axe	CM3a			3.00
	axf			7.5	7.50
Elevador_servo	bba	arl			22.30
	bbb		22.20		22.20
	bbc			2	2.00
	bbd			10	10.00
Suport_potenciòmetres	baa		35.00		35.00
	bab		10.00		10.00
	bac		36.95		36.95
	bad		73.90		73.90
	bae		46.20		46.20
	baf		13.00		13.00
	bag			1.5	1.50
	bah		7.25		7.25
	bai	CM4a			4.00
	baj		5.75		5.75
Grapes_guia_fixe	bca	da			6.00
	bcb			15	15.00
	bcc			1.5	1.50
	bcd			6	6.00
	bce			12.2	12.20
	bcf			37.2	37.20
	bcg			3	3.00
Parets_laterals	bda		847.05		847.05
	bdb	cd			65.00
	bdd	bea			19.30
	bde	cb			15.00
	bdf	byzz			107.90
	bdg	bz			611.95
	bdc	bi			15.00
Ecaires_estructurals	bea			19.3	19.30
	beb			31	31.00
	bec			31	31.00
	bed			1.2	1.20
	bee			5.1	5.10
	bef			7	7.00

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

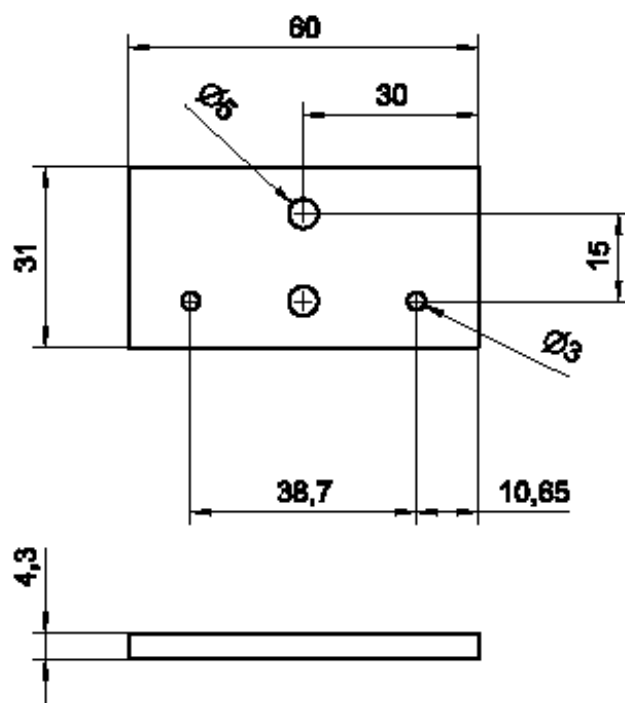
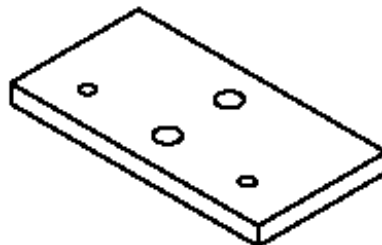
	beg			13.5	13.50
	beh			11	11.00
	bei			7	7.00
Separador_motor	bfa		73.90		73.90
	bfb	aa			4.00
	bfc		50.90		50.90
	bfd		62.40		62.40
	bfe		9.50		9.50
Femella_M4_cega	bha	CM4a			4.00
	bhb			7	7.00
	bhc			4	4.00
	bhd			7	7.00
	bhe			3.5	3.50
	bhf			5	5.00
Femella_M4_autoblocant	bia			7	7.00
	bib	CM4a			4.00
	bic			7	7.00
	bid			3.5	3.50
	bie			1.5	1.50
Insert_fusta-ferro_M4	bja			5	5.00
	bjb	CM4a			4.00
	bjc			10	10.00
Tapa_base_caixa	bka		847.05		847.05
	bkb		695.53		695.53
	bkc			3	3.00
Raspberry_Pi	bla			56.3	56.30
	blb		3.65		3.65
	blc			7	7.00
	bld		6.00		6.00
	ble			89.5	89.50
	blf			51.5	51.50
	blg			25.5	25.50
	blh			6	6.00
	bli			17.2	17.20
	blj			2.7	2.70
	blk			3.5	3.50
	bll			11	11.00
	blm			12	12.00
Camara	bma			25	25.00
	bmb		2.00		2.00
	bmc			21	21.00
	bmd			24	24.00
	bme			12	12.00
	bmf			7.3	7.30
	bmh	bmb		2.2	2.20
					2.00


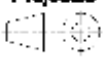
Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

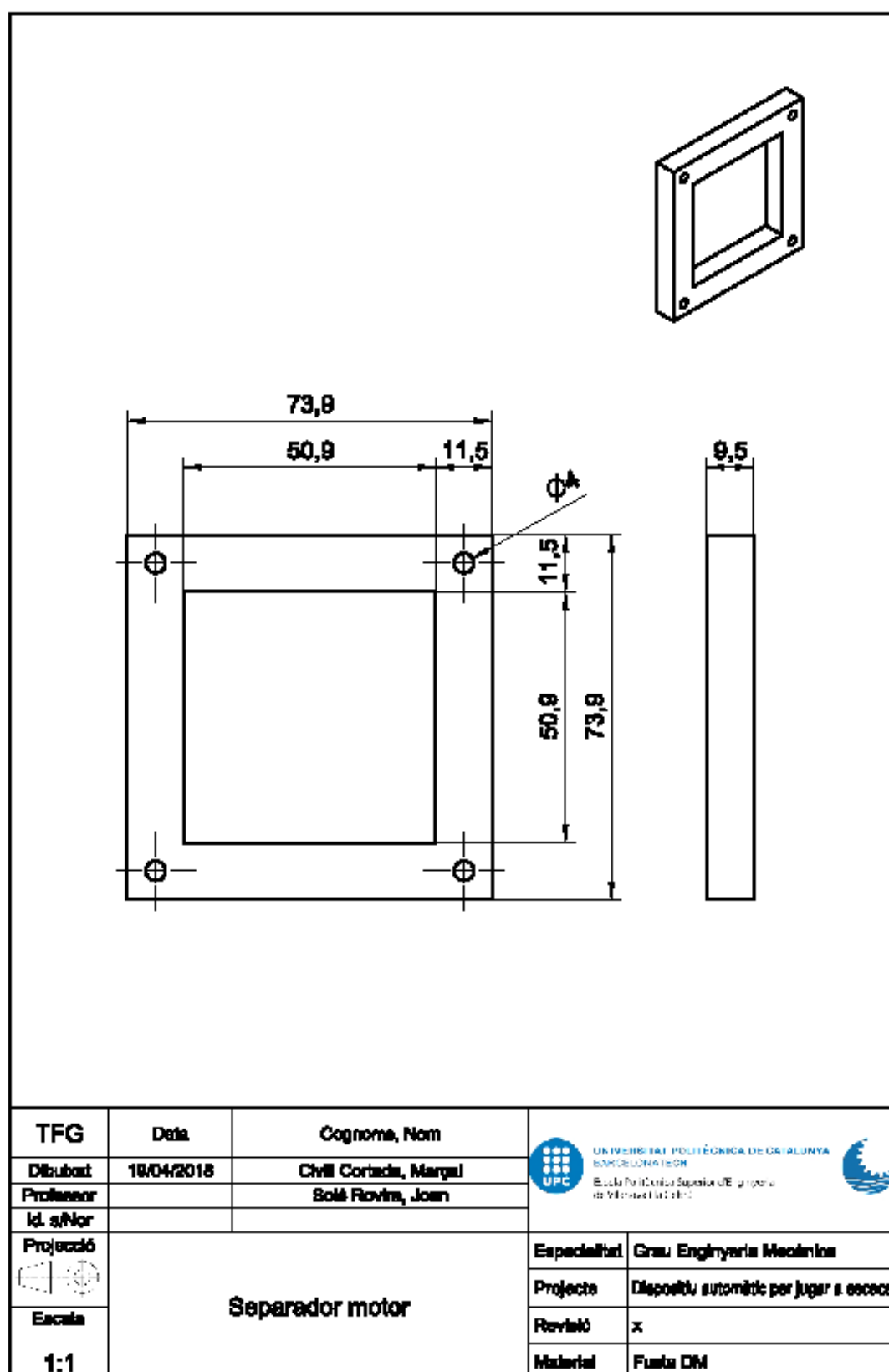
	bmi			5.5	5.50
	bmj			6	6.00
	bm k			2	2.00
	bml			2	2.00
Suport_iman_neodimi	ana		3.00		3.00
	anb			2	2.00
	anc			4	4.00
	and	aya			6.00
	ane		15.00		15.00
	anf			0.6	0.60
	ang	aya			6.00
	anh			8	8.00
Casquillo_sup_corretja	bga			6	6.00
	bg b	CM4a			4.00
	bg c		7.80		7.80
Escaires_sup_guia_fixe	aza	bea			19.30
	azb	bef			7.00
	azc	beg			13.50
	azd	bee			5.10
	aze			25	25.00
	azf	bec			31.00
	azg	bed			1.20
	azh	azb			7.00
	azi	azc			13.50
	azj	beh			11.00
	azk	bei			7.00
Escaires_lat_sup_cam	bna			16.5	16.50
	bnb			50	50.00
	bnc			50	50.00
	bnd			2	2.00
	bne			5	5.00
	bnf			7	7.00
	bng			14	14.00
	bnh			26	26.00
	bni			8	8.00

PLÀNOLS

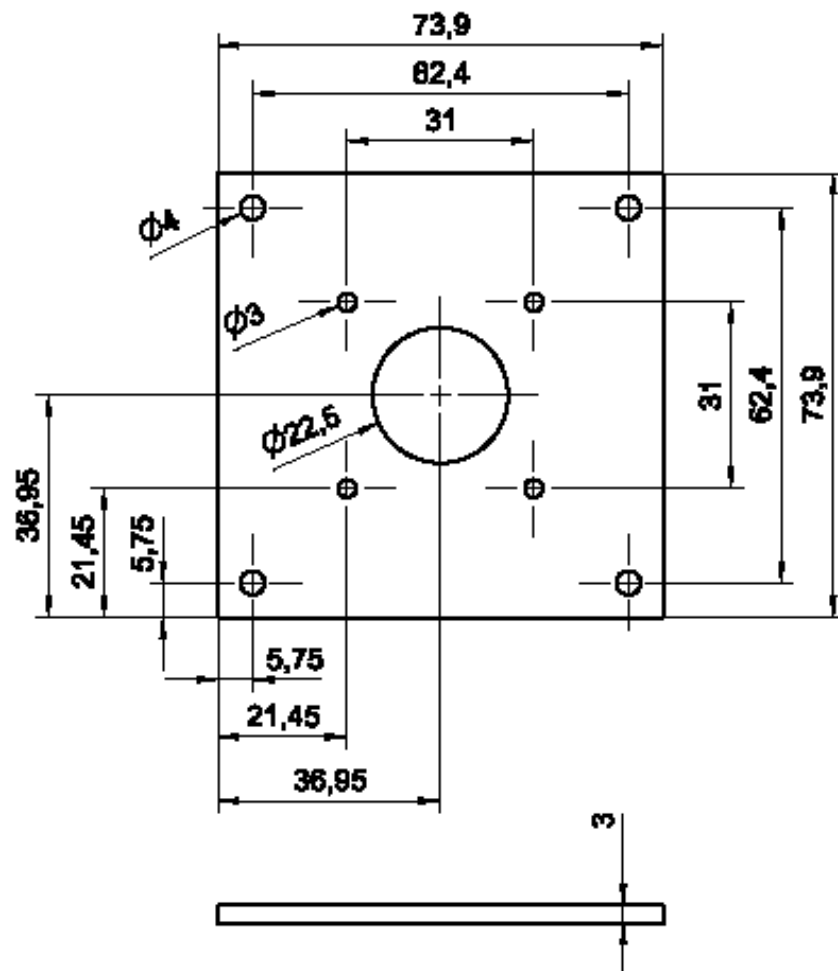
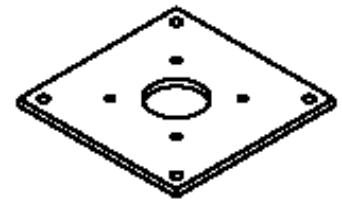
Nota: Tots els forats són passants.






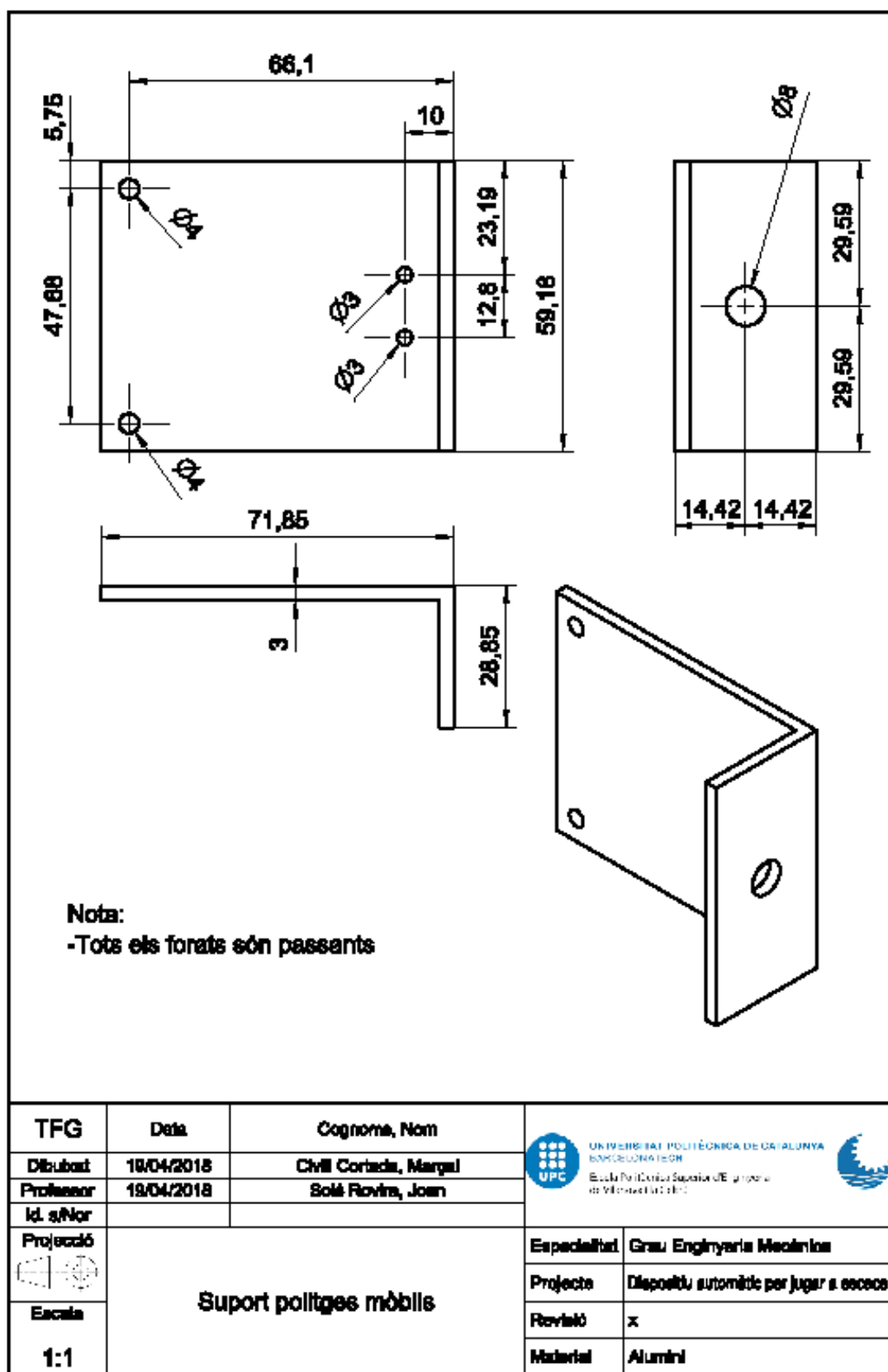
TFG	Data	Cognome, Nom	 UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONA TECH Escola Politècnica Superior d'Enginyeria de Màquines i Tèxtils	
Dibuixat	20/05/2018	Civill Cortada, Marçal		
Professor		Solà Rovira, Joan		
Id. s/Nor				
Projectió			Especialitat	Grau Enginyeria Mecànica
Escala			Projecte	Dispositiu automàtic per jugar a escacs
1:1			Revisió	x
			Material	Alumini + compost



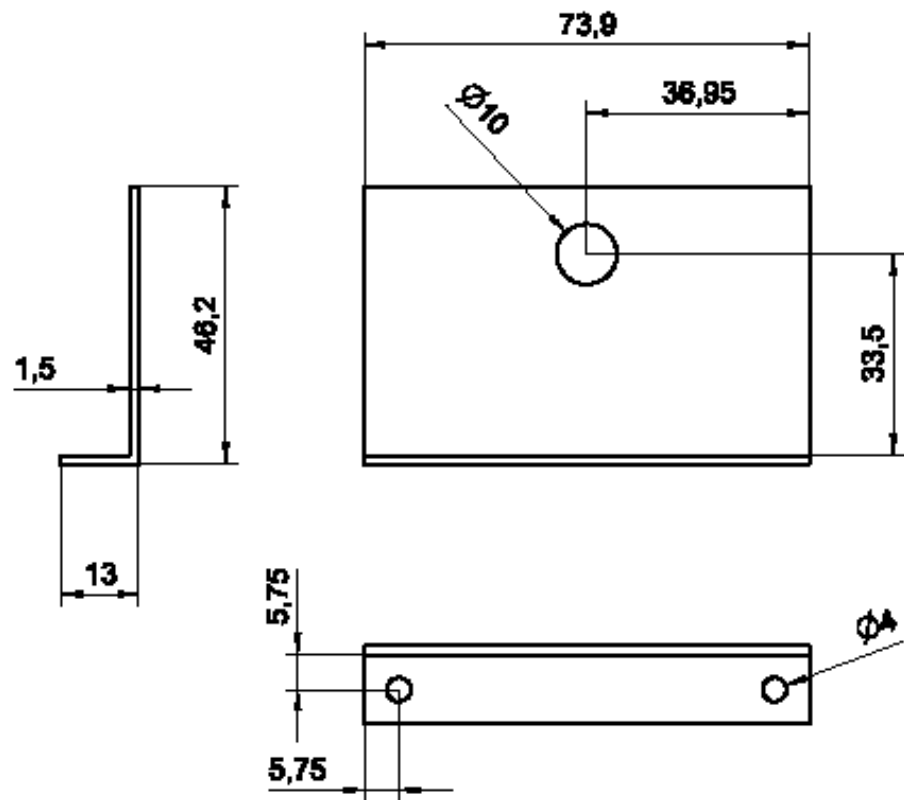
Nota:
- Tots els forats són passants




TFG	Data	Cognome, Nom	 UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONA TECH Escola Politècnica Superior d'Enginyeria de Màquines i Tèxtils 	
Dibuixat	18/04/2018	Civill Cortada, Marçal		
Professor		Solà Rovira, Joan		
Id. s/Nor				
Projectió	Suport motor		Especialitat	Gradu Enginyeria Mecànica
			Projecte	Dispositiu automàtic per jugar a escacs
Escala			Revisió	x
1:1			Material	Alumini

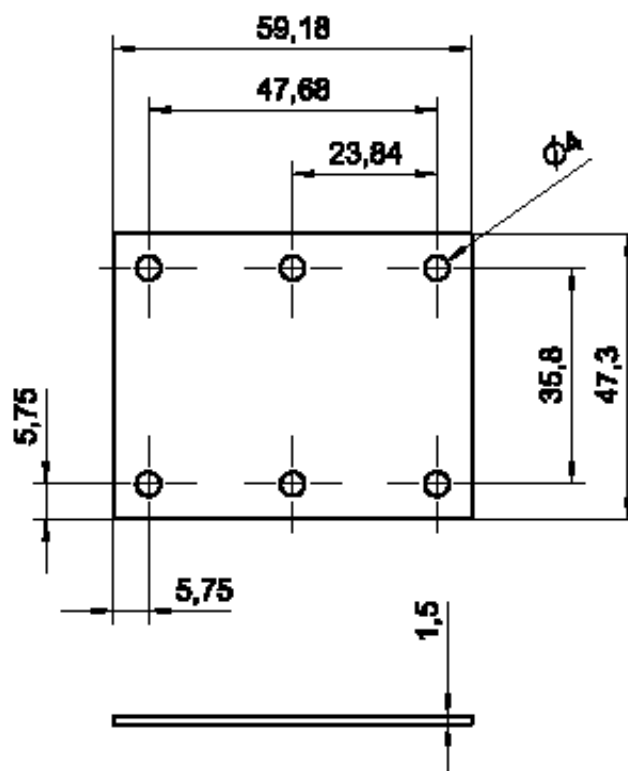
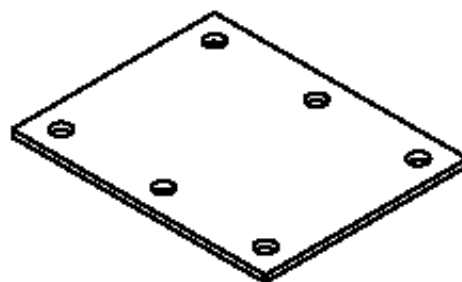




Nota: Tots els forats son passants.

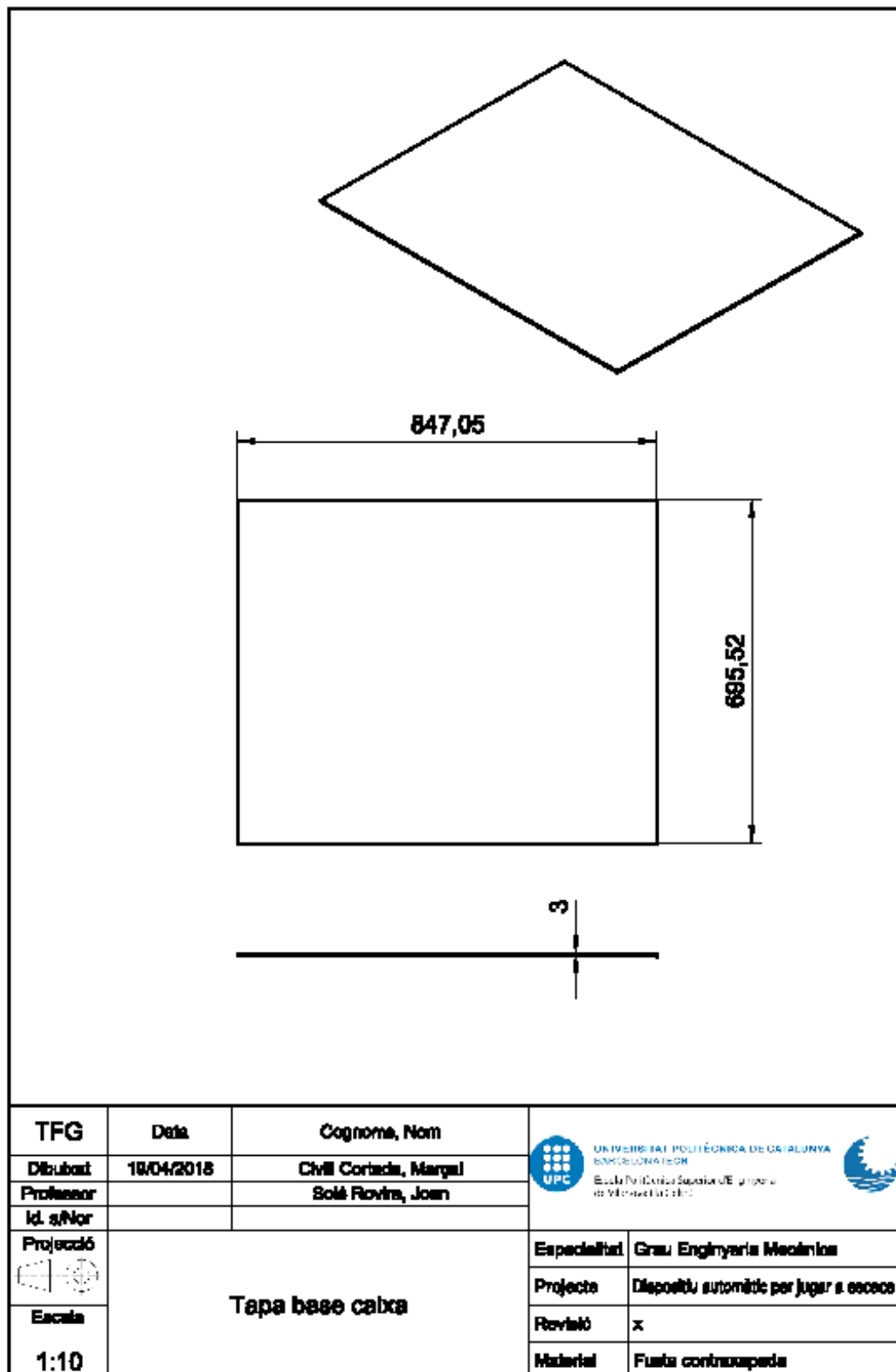


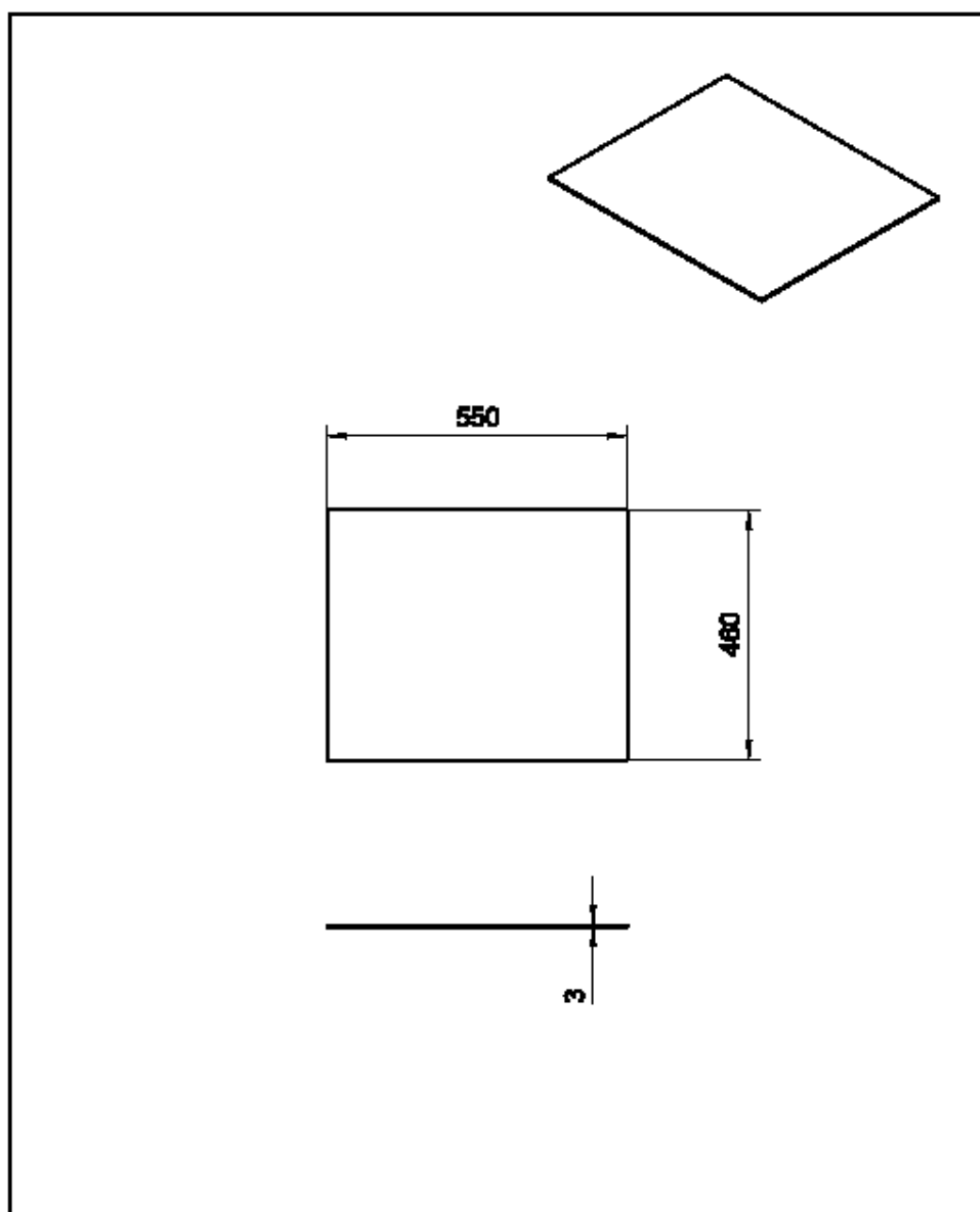
TFG	Data	Cognome, Nom	 UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONA TECH Escola Politècnica Superior d'Enginyeria de Màquines i Tèxtils	
Dibutxat	18/04/2018	Civill Cortada, Marçal		
Professor		Solà Rovira, Joan		
Id. s/Nor				
Projectat	Suport potenciómetre		Especialitat	Grau Enginyeria Mecànica
Escales			Projecte	Dispositiu automàtic per jugar a escacs
1:1			Revisió	x
			Material	Alumini



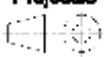
Nota: Tots els forats son passants.

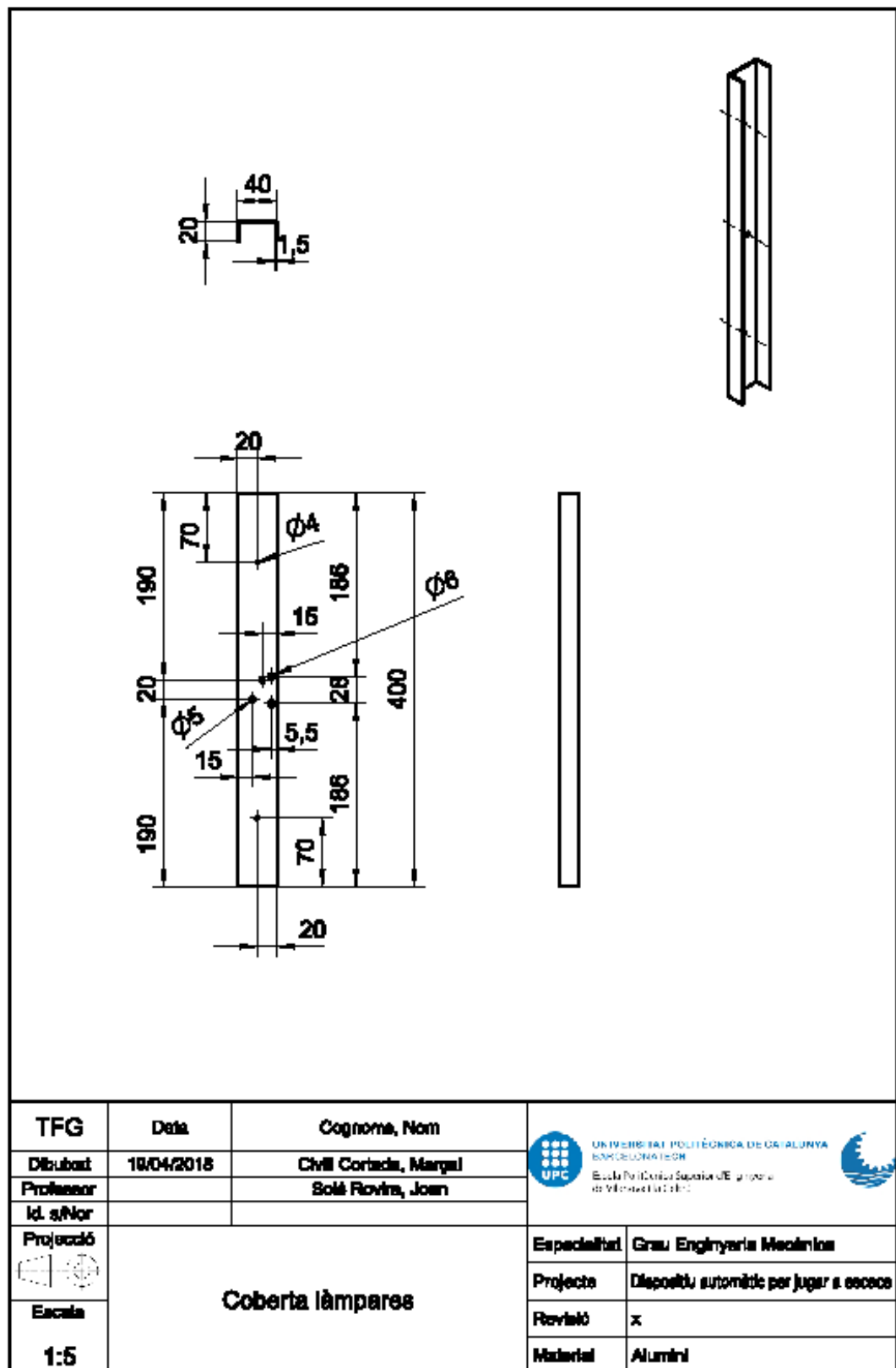


TFG	Data	Cognoma, Nom	 UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONA TECH Escola Politècnica Superior d'Enginyeria de Màquines i Tèxtils	
Dibutxat	10/04/2018	Civill Cortada, Marçal		
Professor		Saló Rovins, Joan		
Id. s/Nor				
Projecte	Suport servomotor		Especialitat	Grau Enginyeria Mecànica
			Projecte	Dispositiu automàtic per jugar a escacs
Escales			Revisió	x
1:1			Material	Alumini

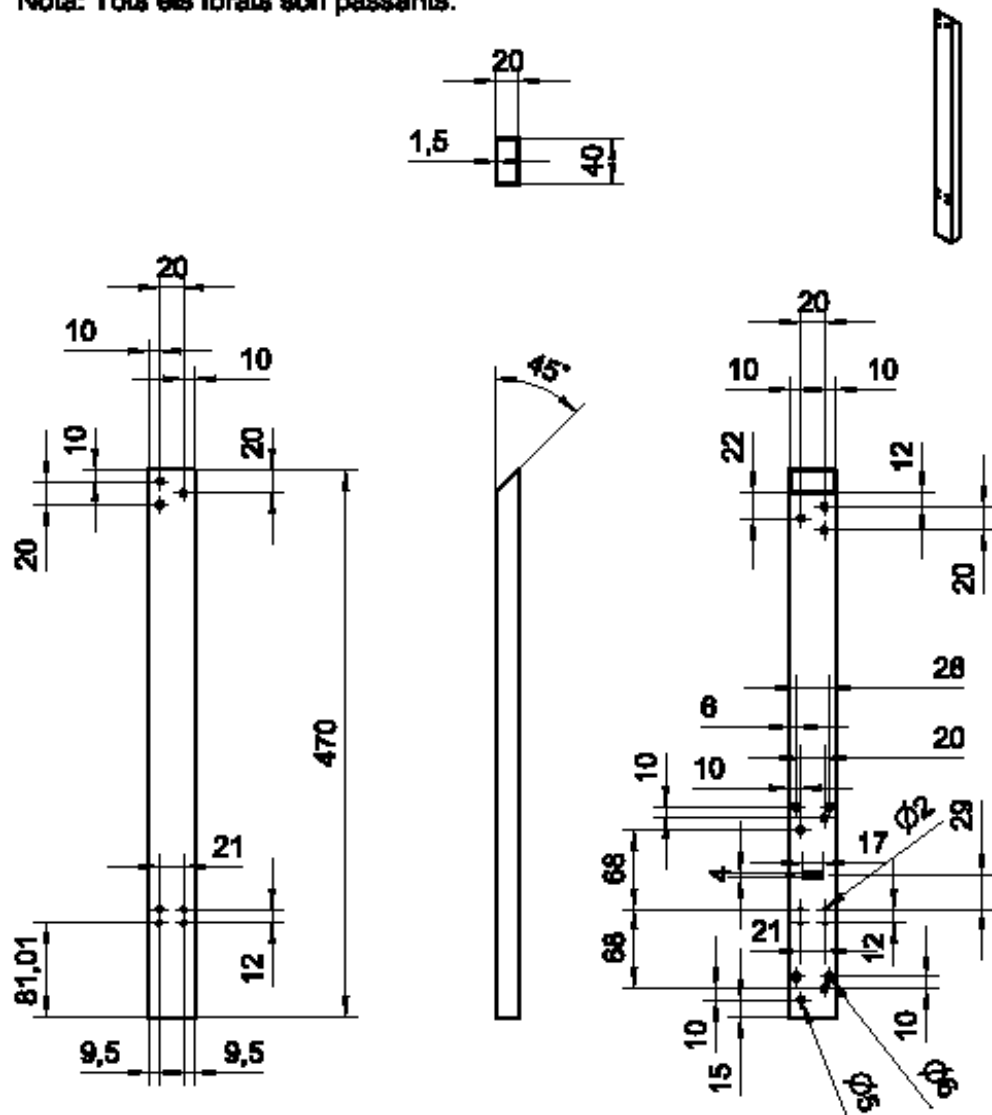





TFG	Data	Cognoma, Nom	 UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONA TECH Escola Politècnica Superior d'Enginyeria de Màquines i Tèxtils 	
Dibutxat	18/04/2018	Civill Cortada, Marçal		
Professor		Solé Rovira, Joan		
Id. s/Nor				
Projectat 	Tauler		Especialitat	Gradu Enginyeria Mecànica
Escala 1:10			Projecte	Dispositiu automàtic per jugar a escacs
			Revisió	x
			Material	Fusta contraspada

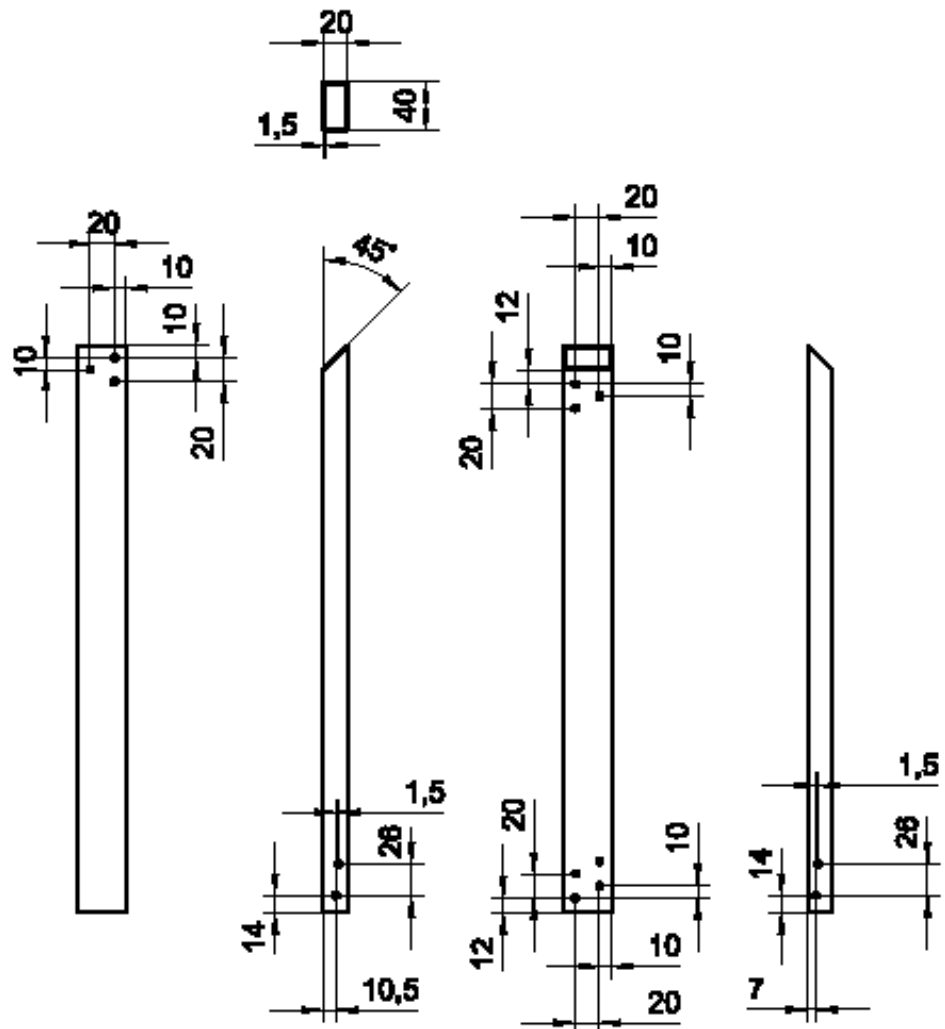




Nota: Tots els forats són passants.



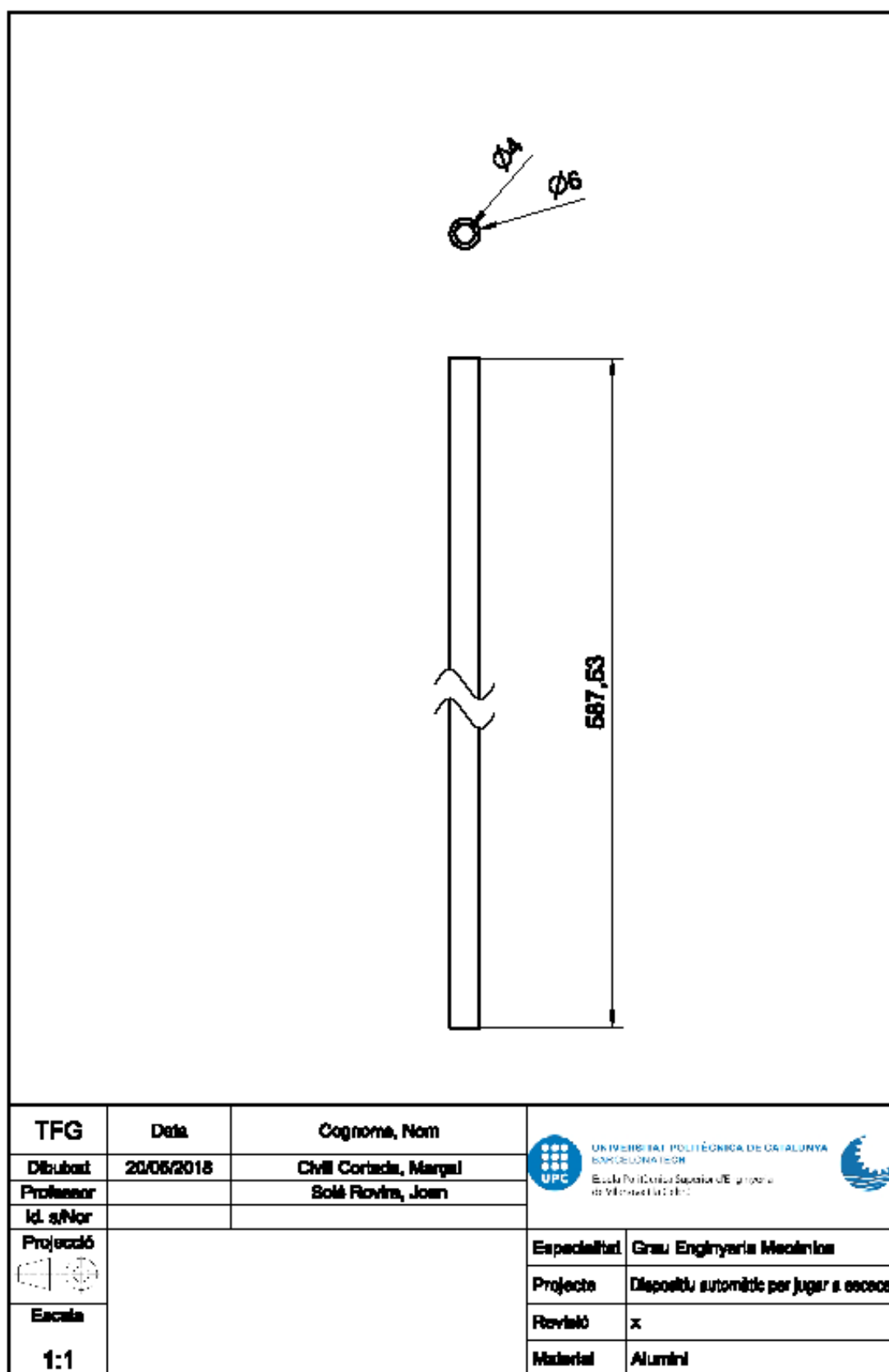
TFG	Data	Cognome, Nom	 UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONA TECH Escola Politècnica Superior d'Enginyeria de Màquines i Tèxtils	
Dibuixat	18/04/2018	Civill Cortada, Marçal		
Professor		Solà Rovira, Joan		
Id. s/Nor				
Projectat	Estructura horitzontal càmera		Especialitat	Grau Enginyeria Mecànica
Escala			Projecte	Dispositiu automàtic per jugar a escacs
1:5			Revisió	x
			Material	Alumini

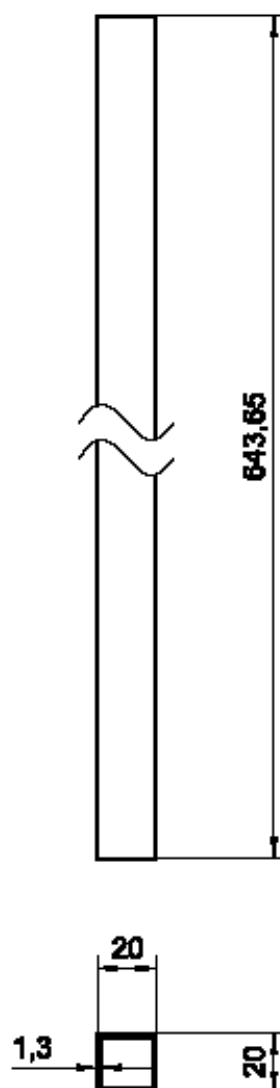
Nota: Tots els forats són passants.



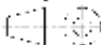


TFG	Data	Cognome, Nom	 UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONA TECH Escola Politècnica Superior d'Enginyeria de Màquines i Tèxtils	
Dibutxat	16/04/2018	Civill Cortada, Marçal		
Professor		Saló Rovira, Joan		
Id. s/Nor				
Projecte			Especialitat	Grau Enginyeria Mecànica
Escales			Projecte	Dispositiu automàtic per jugar a escacs
1:5			Revisió	x
			Material	Alumini

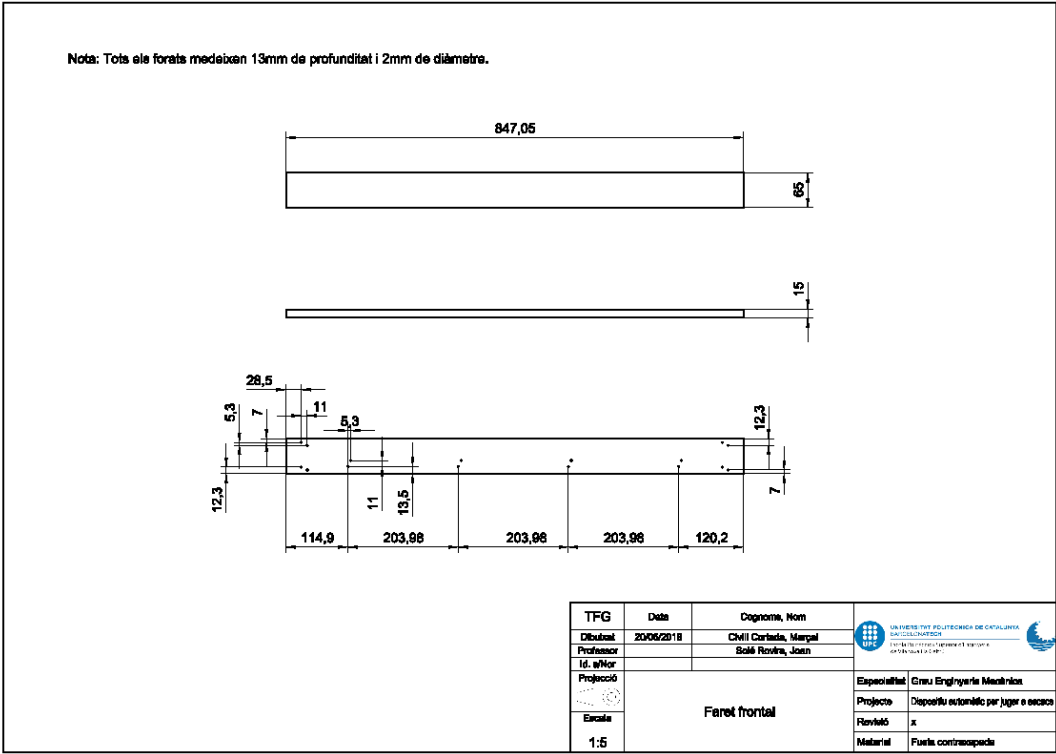
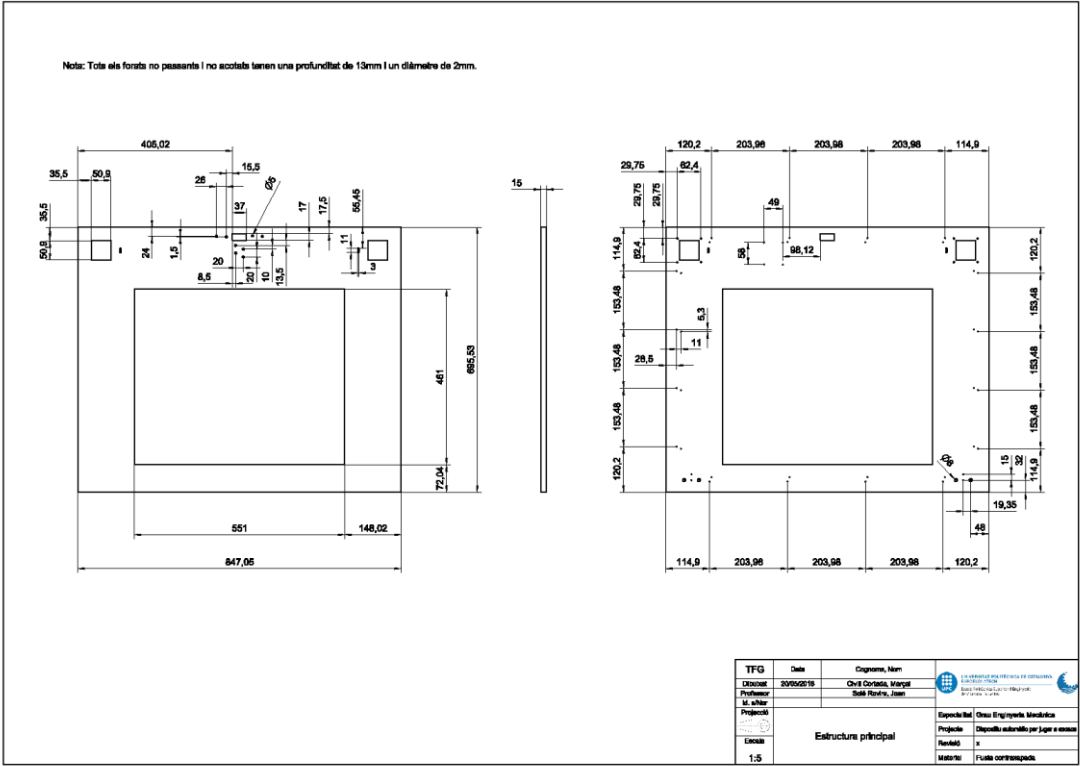
Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada



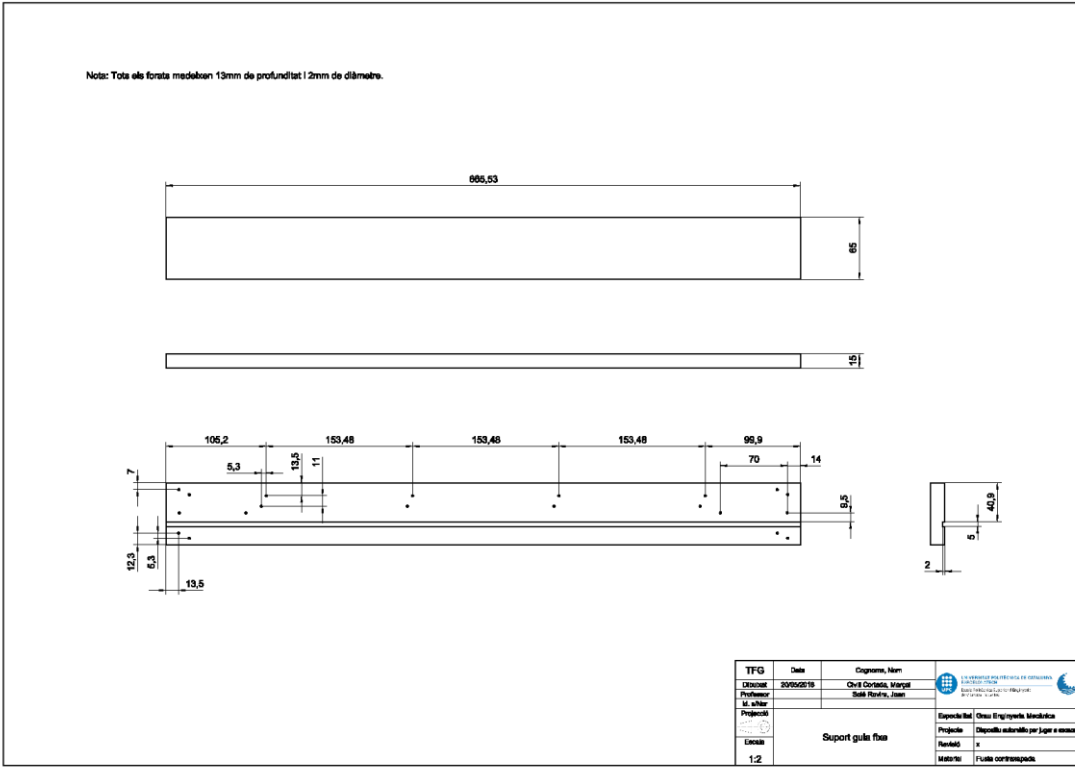
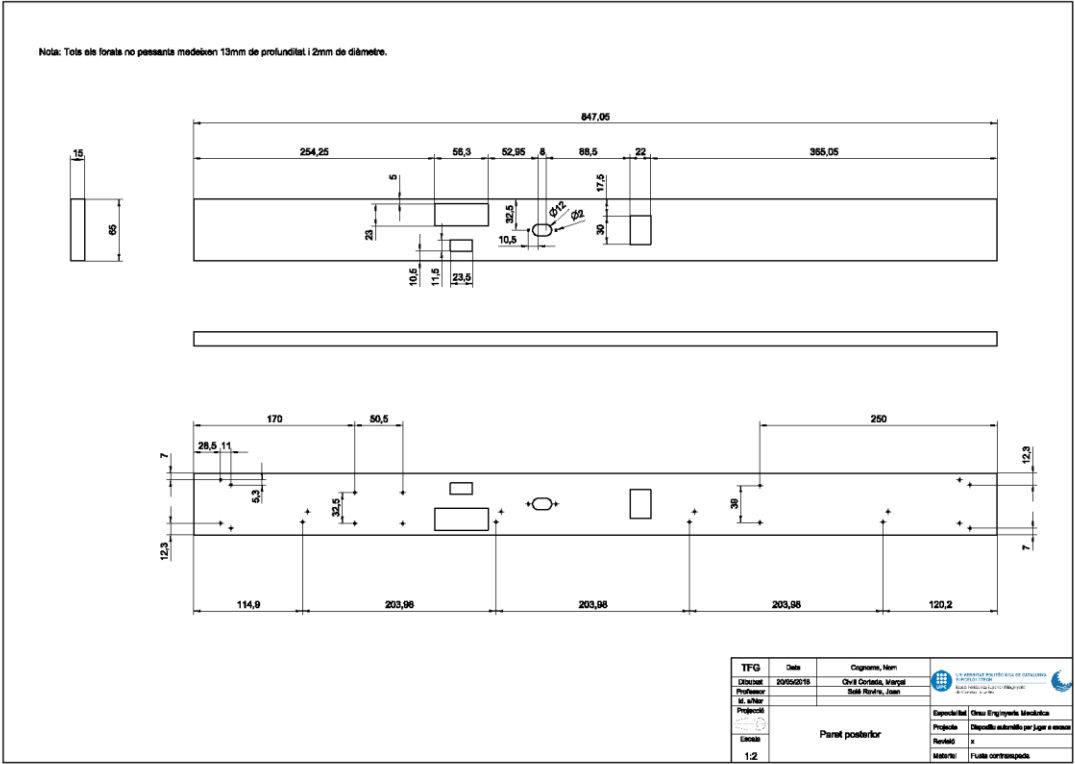


TFG	Data	Cognoma, Nom	 UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONA TECH Escola Politècnica Superior d'Enginyeria de Màquines i Tèxtils	
Dibuixat	18/04/2018	Civill Cortada, Marçal		
Professor		Solà Rovins, Joan		
Id. s/Nor				
Projecció 	Gua mòbil		Especialitat	Grau Enginyeria Mecànica
Escala 1:2			Projecte	Dispositiu automàtic per jugar a escacs
			Revisió	x
			Material	Alumini

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada



Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada



FUNCIONS D'ELABORACIÓ PRÒPIA UTILITZADES EN EL PROGRAMA

FUNCIO PER ENQUADRAR LA IMATGE

```
1. def encuadrar(img, llista_coord):
2.     """
3.     img = imatge d'entrada
4.     [[x1,y1],[x2,y2],[x3,y3],[x4,y4]] = els 4 pixels de la imatge que defineixen
5.     l'àrea que es vol encuadrar (de dreta a esquerra i de dalt a baix)
6.     [xmax,ymax] = dimensions de la imatge final encuadrada (en píxels)
7.     """
8.
9.     global xmax
10.    x1,y1,x2,y2,x3,y3,x4,y4 = coord_encuadrar
11.    pts1 = np.float32([[x1,y1],[x2,y2],[x3,y3],[x4,y4]])
12.    pts2 = np.float32([[0,0],[xmax,0],[0,xmax],[xmax,xmax]])
13.
14.    M = cv2.getPerspectiveTransform(pts1,pts2)
15.    enc = cv2.warpPerspective(img,M,(xmax,xmax))
16.
17.    return enc
```

La imatge que ha captat la càmera no està perfectament enquadrada en el taulell, té certa perspectiva i zones que surten de l'àrea de joc. Per a l'anàlisi d'imatge es requereix que en la foto només aparegui l'àrea de joc sense perspectiva.

Per fer-ho s'utilitzen les funcions d'OpenCV "cv2.getPerspectiveTransform" i "cv2.warpPerspective" que actuen de la forma següent:

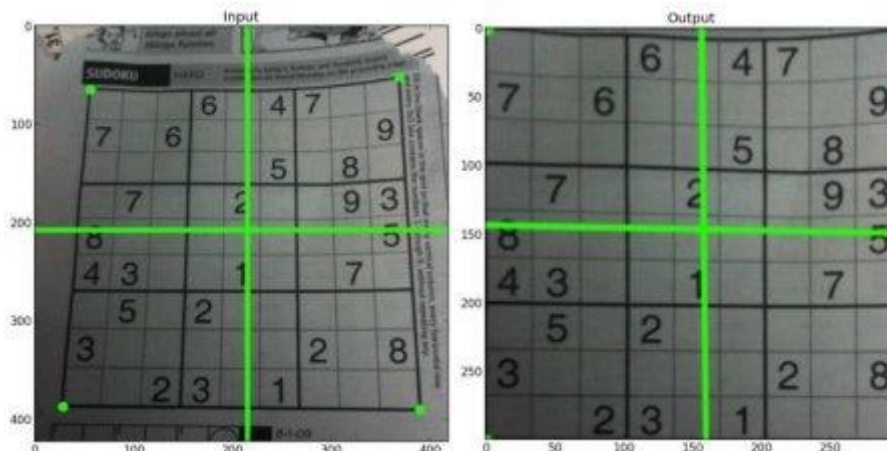


Figura 160. Transformació de perspectiva. (Font: cv2.warpPerspective)

Per utilitzar aquesta funció s'ha d'introduir la imatge que es vol transformar juntament amb les coordenades (en píxels) dels vèrtexs de la imatge inicial que es vol encuadrar i per altra banda les coordenades en píxels on han d'anar cada un dels punts de l'àrea enquadrada en la imatge final.

FUNCIÓ PER ROTAR LA IMATGE

```
1. def r180(img):
2.     """
3.     Rotar la imatge 180º
4.     """
5.     # Rotar la imatge del taulell 180º, ja que la càmera enfoca des del costat
    oposat al jugador
6.     if img[0][0].shape == ():
7.         rows,cols = img.shape
8.     else:
9.         rows,cols,extra = img.shape
10.
11.     M = cv2.getRotationMatrix2D((cols/2,rows/2),180,1)
12.     img180 = cv2.warpAffine(img,M,(cols,rows))
13.
14.     return img180
```

Com que la càmera esta col·locada de manera inversa a la posició del jugador, la foto que capta també esta invertida. Per a que el programa funcioni es requereix que només hi hagi una sola orientació del taulell i s'ha escollit la perspectiva que veu l'humà. Aquesta funció gira la imatge 180 graus utilitzant les funcions d'OpenCV "cv2.getRotationMatrix2D" i "cv2.warpAffine".

FUNCIÓ PER DETECTAR LES PECES

```
1. def cercles(img):
2.     """
3.     Detectar cercles
4.     """
5.     img = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY) # Passa la imatge a escala de gr
    isos
6.
7.     img = cv2.medianBlur(img,5) # Suavitzar la imatge
8.
9.     # Detectar cercles
10.    cimg = cv2.cvtColor(img,cv2.COLOR_GRAY2BGR)
11.    circles = cv2.HoughCircles(img, cv2.HOUGH_GRADIENT,1, 20, param1=40, param2
    =23, minRadius=22, maxRadius=32)
12.    #circles = cv2.HoughCircles(img, cv2.HOUGH_GRADIENT,1, 20, param1=40, param
    2=23, minRadius=22, maxRadius=32) # Millor comprovat
13.    circles = circles[0,:,:] # S'elimina un nivell de matriu innecesari creat p
    er Hough Circles
14.    circles = np.uint16(np.around(circles)) # S'arrodoneixen els valors de coor
    denades i radis
```

En aquesta funció s'agafa la imatge del taulell enquadrada i girada 180º i es segueix el següent procediment:

Es transforma la imatge en color a escala de grisos ja que la funció de detecció de cercles només funciona en imatges en blanc i negre. Per fer-ho s'utilitza la funció d'OpenCV "cv2.cvtColor" i se l'hi aplica la característica "cv2.COLOR_BGR2GRAY" amb aquestes funcions es converteix la imatge inicialment en format BGR (Blue Green Red) a escala de grisos.

Seguidament es suavitza la imatge per reduir els efectes del soroll i així obtenir menys falsos positius. Per fer-ho s'utilitza la funció "cv2.medianBlur" que actua de la següent manera:

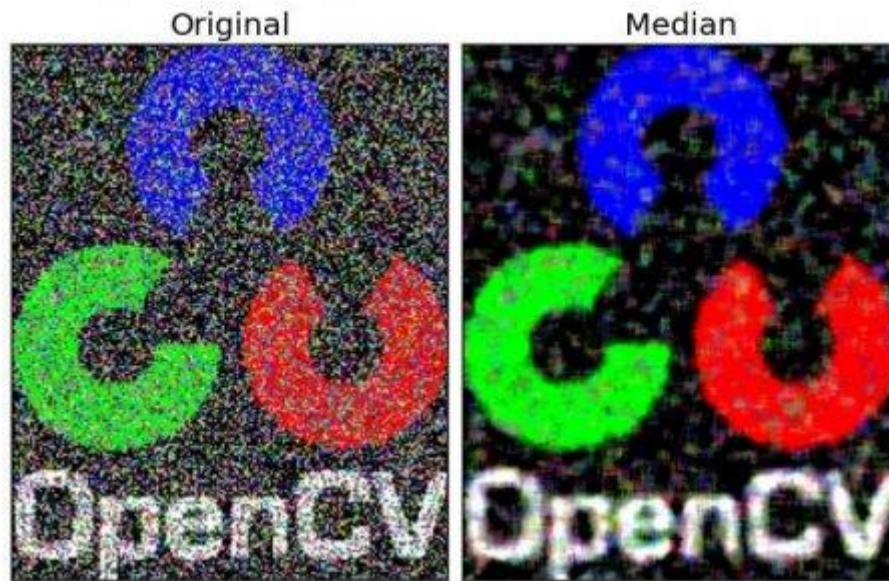


Figura 120. Aplicació del "Median Blurring" (Font: <https://docs.opencv.org>)

Un cop suavitzada la imatge es procedeix a detectar els cercles de la imatge. L'objectiu és detectar les bases de les peces des de la perspectiva vertical i així saber en quins escacs hi ha peces i quins estan buits. Per detectar els cercles s'utilitza la funció d'OpenCV "*cv2.HoughCircles*" en aquesta funció se l'hi han d'introduir diversos paràmetres com ara els radis mínim i màxim dels cercles que es volen detectar entre d'altres. Aquests paràmetres s'han d'ajustar fent proves fins a trobar els més òptims.

Aquesta funció crea una matriu de tres columnes corresponents a: (radi del cercle, posició x del centre del cercle (en píxels), posició y del centre del cercle (en píxels)). El nombre de files correspondrà al nombre de cercles detectats ja que conté un cercle per fila.


```
1. ##### FILTRAR CERCLES ##### (Disenyat per eliminar els caps del rei i la reina)
2.
3.     global xymax
4.     global amplada_escac
5.
6.     n_cercle = 0
7.
8.     cercles_remake = np.empty((circles.shape[0],7), dtype=int)
9.
10.    for x0, y0, r in circles:
11.
12.        tP1_posicio_x = ((x0/xymax)*8)%1 # Proporció entre la distancia des del
            centre de la peça al costat esquerre de l'escac i l'amplada total de l'escac (
            expresat en tant per 1)
13.        tP1_posicio_y = ((y0/xymax)*8)%1 # Proporció entre la distancia des del
            centre de la peça al costat superior de l'escac i l'altura total de l'escac (e
            xpresat en tant per 1)
14.
15.        dt = tP1_posicio_y * amplada_escac # Distancia des del centre del cercle
            fins a l'aresta superior de l'escac (els escacs medeixen 100 x 100 píxels) (d
            t = distancia "top")
16.        dr = (1 - tP1_posicio_x)*amplada_escac # Idem fins a l'aresta dreta (dr
            = distancia "right")
17.        db = (1 - tP1_posicio_y)*amplada_escac #Idem fins a baix (db = distanci
            a "bottom")
18.        dl = tP1_posicio_x * amplada_escac # Idem fins a l'esquerra (dt = dista
            ncia "top")
19.
20.        limit = (centratge * r) # Calsevol cercle detectat que tingui un 20% d
            e la seva amplada fora del seu escac, serà descartat.
```

Quan es detecten cercles amb la funció comentada anteriorment s'obtenen els cercles que s'estan buscant (els de les bases) juntament amb alguns cercles que no interessen (normalment els caps del rei i la reina) i altres cercles que anomenarem "falsos positius" que corresponen a cercles que l'algoritme detecta en llocs on un humà no hi veuria cap cercle. Això passa normalment a causa de les ombres de les peces sobre el taulell i per això es extremadament important una bona il·luminació vertical.

Per eliminar aquests cercles que no corresponen a les bases de les peces s'apliquen una sèrie de comparacions i operacions matemàtiques derivades de raonaments lògics, com per exemple:

S'eliminaran tots els cercles que estiguin continguts en menys d'un 80% de seu diàmetre dins d'un escac.

Això és útil ja que les ombres acostumen a caure entre escacs. Per al cas dels caps del rei i la reina també serveix quan estan en escacs de la perifèria, ja que al ser figures altes la perspectiva fa que aquests s'acostumin a veure en els límits entre escacs.

```
1. if dt < limit or dr < limit or db < limit or dl < limit:
2.     # draw the outer circle
3.     cv2.circle(cimg,(x0,y0),r,(0,0,255),2)
4.     # draw the center of the circle
5.     cv2.circle(cimg,(x0,y0),2,(0,0,255),3)
6.
7.     cercles_remake = np.delete(cercles_remake, n_cercle, 0)
8.
9.     else:
10.        ##### DIBUIXAR ELS CERCLES #####
11.
12.        # draw the outer circle
13.        cv2.circle(cimg,(x0,y0),r,(255,0,0),2)
14.        # draw the center of the circle
15.        cv2.circle(cimg,(x0,y0),2,(255,0,0),3)
16.
17.        ##### GUARDAR LES DADES A LA NOVA LLISTA DE CERCLES #####
18.
19.        cercles_remake[n_cercle][0] = int(x0)
20.        cercles_remake[n_cercle][1] = int(y0)
21.        cercles_remake[n_cercle][2] = int(r)
22.        cercles_remake[n_cercle][3] = int(dt)
23.        cercles_remake[n_cercle][4] = int(dr)
24.        cercles_remake[n_cercle][5] = int(db)
25.        cercles_remake[n_cercle][6] = int(dl)
26.
27.        n_cercle += 1 # Es passa al següent cercle de la llista
28.
29.
30.    return cercles_remake, cimg # Retorna una llista amb els cercles trobats i
    una imatge amb els cercles dibuizats
```

Els cercles que no passen la prova són eliminats de la llista i es dibuixen en vermell sobre la imatge.

Els cercles que passen la prova es mantenen dins la llista i es dibuixen en verd sobre la imatge.

FUNCIÓ PER DETECTAR EL COLOR

```
1. def posicio_color(img, pix_cons_img, cercles_llista):
2.     """
3.     Detecta la mitjana de color en una area quadrada circumscripita en un cercle
       donat
4.
5.     Només admet imatges en color (BGR)
6.
7.     img = imatge (en color BGR)
8.     cons = imatge on es pintaran de color blanc els píxels consultats
9.     c_llista = llista que conté les coordenades del centre i el radi de diversos
       cercles
10.
11.     P_BN_llista = llista on es guardaran: nº escac x (0-
       >7, d'esquerra a dreta), nº escac y (0->7, de dalt a baix),
12.     """
13.
14.     P_BN_llista = np.empty(cercles_llista.shape[0], dtype = [('cx',int), ('cy',
       int), ('r',int), ('px',int), ('py',int), ('dt',int), ('dr',int), ('db',int), ('
       dl',int), ('mB', int), ('mG', int), ('mR',int), ('c',int)])
15.
16.     n_cercle = 0 # Indica la posició dins de la llista del cercle analitzat en
       cada cicle del bucle
```

Cada cercle correspon a una peça vista des de dalt. Coneixent les posicions però no el color de les peces falta informació per poder reconèixer els moviments.

Per conèixer el color de les peces detectades es segueix el següent procediment:

Es crea una nova llista on es guardaran les dades del cercle i les mitjanes de color.

```
1. for x0,y0,r,dt,dr,db,dl in cercles_llista:
2.     """
3.     x0 = coordenada x (en píxels) del centre de la circumferència
4.     y0 = coordenada y (en píxels) del centre de la circumferència
5.     r = radi de la circumferència
6.     """
7.     ##### COLPIA DE COORDENADES I RADIS #####
8.
9.     P_BN_llista[n_cercle]['cx'] = int(x0)
10.    P_BN_llista[n_cercle]['cy'] = int(y0)
11.    P_BN_llista[n_cercle]['r'] = int(r)
12.
13.    ##### COLPIA DE DISTANCIES CENTRE-ARESTES #####
14.
15.    P_BN_llista[n_cercle]['dt'] = int(dt)
16.    P_BN_llista[n_cercle]['dr'] = int(dr)
17.    P_BN_llista[n_cercle]['db'] = int(db)
18.    P_BN_llista[n_cercle]['dl'] = int(dl)
```

Es copien les dades dels cercles de la anterior llista a la nova.

```
1. ##### Càlcul de posició #####
2. """
3.     Calcular la posició de la peça en el taulell (en escacs) a partir de la
       posició (en píxels) del centre de la base
4.     px -> columna de la matriu (d'esquerra a dreta entre 0 i 7)
5.     py -> fila de la matriu (de dalt a baix entre 0 i 7)
6.     """
7.     px = math.floor((x0/ymax)*8) # Cercles[i][0] = posició x del centre d
       e la circumferència (en píxels), 400 = nº píxels totals imatge, 4 = nº total de
       columnes
8.     py = math.floor((y0/ymax)*8) # (es divideix la posició en píxels ent
       re els píxels totals i s'obté la posició en %. Al multiplicar per 4 s'obté la
       posició en %4 que es el nº de columnes totals. Al arrodonir cap avall amb la fu
       nció floor() s'obtenen valors entre 0 i 3 segons la posició de les peces.
9.
10.    P_BN_llista[n_cercle]['px'] = int(px)
11.    P_BN_llista[n_cercle]['py'] = int(py)
```

Es calcula la posició de cada peça en el taulell (en escacs) a partir de les coordenades del centre (en píxels) i es guarden a la nova llista.

```
1. ##### Càlcul del color mitjà #####
2. """
3.     Detecta la mitjana de color en una àrea quadrada circumscrita en un cer
       cle donat
4.     """
5.     d = int(math.floor(r * 0.7071)) # Distància des del centre la la circ
       umferència fins a les cares del quadrat analitzat (0,7071 = arrel(0,5))
6.
7.     xi = int(x0 - d) #píxels de referència del quadrat sobre la imatge orig
       inal
8.     yi = int(y0 - d)
9.     xf = int(x0 + d)
10.    yf = int(y0 + d)
11.
12.    quadrat = img[yi:yf,xi:xf,:] # Imatge quadrada inscrita en el cercle
13.
14.    suma = np.array([np.sum(quadrat[:, :, 0]), np.sum(quadrat[:, :, 1]), np.sum
       (quadrat[:, :, 2])]) #Si interessen la resta de colors: suma = np.sum(quadrat
      [:, :, 2])
15.    n_pixels = quadrat.shape[0]*quadrat.shape[1]
16.    mitjana = (np.true_divide(suma,n_pixels))*(-1) # Multiplicant per -
       1, quan s'ordeni la matriu les blanques apareixeran primer
17.
18.    P_BN_llista[n_cercle]['mB'] = int(mitjana[0]) # Blue (residual)
19.    P_BN_llista[n_cercle]['mG'] = int(mitjana[1]) # Green (secundari)
20.    P_BN_llista[n_cercle]['mR'] = int(mitjana[2]) # Red (principal)
21.
22.    pix_cons_img[yi:yf,xi:xf,:] = 255 # Marcar els píxels consultats en
       blanc
23.
24.    ##### FINAL DEL CICLE #####
25.
26.    n_cercle += 1
```

Per detectar si la peça es blanca o negra s'analitza el color mitjà en una àrea quadrada circumscrita en cada un dels cercles de la llista. Com que la imatge està en format BGR les tres components del color es guarden per separat, ja que després serà més fàcil distingir entre "blanc" i "negre" si els tres colors estan per separat.

En les peces blanques mostren un valor de color vermell molt alt en comparació a les peces negres, mitjançant aquest valor es fa la distinció.

```
1. ##### ASSIGNACIÓ BLANC/NEGRE #####
2. """
3.     Es conten les peces blanques dins de t_ant, s'ordena la lista de peces "P_B
   N_llista" de valor més alt de color vermell a més baix,
4.     i finalment s'assigna color blanc a les "n" primeres peces de la llista, on
   "n" es el nombre de peces blanques presents en t_ant.
5.
6.     Aquest sistema (que l'hi diré "sistema amb tall variable") és més efectiu q
   ue determinar un valor de tall entre blanc i negre,
7.     ja que amb les condicions d'il.luminació aquest pot variar:
8.
9.     En un sistema amb un valor de tall fix:
10.    Si la imatge s'enfosqueix, algunes peces blanques es poden detectar com a n
   egres i si passa al contrari, si està molt il·luminada, algunes peces
11.    negres es detectarien com a blanques.
12.
13.    En un sistema amb tall variable, com es el cas:
14.    Encara que la il:luminació varii, variarà per totes les peces per igual, i
   per tant, les peces blanques sempre es veuran més blanques que les negres,
15.    encara que la seva mitjana de color varii. Per tant, si s'ordenen de més cl
   ar a més fosc i es coneix el nombre de peces blanques que s'han de detectar,
16.    es pot saber quines son blanques i quines son negres sense importar el valo
   r exacte del seu color.
17.    """
18.    global t_ant # En aquest moment t_ant emmagatzema la posició de les peces d
   esprés del torn de la màquina
19.
20.    M_blanques = np.array(t_ant==2) # Es crea una matriu True/False on True = p
   eça blanca i False = peça negra
21.
22.    n_blanques = np.count_nonzero(M_blanques) # Es conta la quantitat de "True"
   (peces blanques)
23.
24.    P_BN_llista = np.sort(P_BN_llista, axis=0, kind='quicksort', order=['mR', '
   mG'])
25.
26.    P_BN_llista[0:(n_blanques)]['c'] = 2 # Assignació color blanc a les n prime
   res peces de la llista ordenada de blanc a negre (n = nº peces blanques)
27.    P_BN_llista[n_blanques:(P_BN_llista.shape[0])]['c'] = 3 # Assignació color
   negre a la resta de peces
28.
29.
30.    return P_BN_llista, pix_cons_img # Color mitjà dels pixels consultats // I
   matge amb els pixels marcats
```

Per trobar quines peces són blanques i quines negres es fa el següent:

Primerament s'agafa la matriu on s'emmagatzemen les posicions i colors de les peces en el taulell anterior i es compten la quantitat de peces blanques. Com que l'humà juga amb les blanques és impossible que cap peça blanca hagi mort en el transcurs d'aquest torn, per tant n'hi haurà la mateixa quantitat.

Seguidament s'ordena la llista de cercles posicions i colors en ordre decreixent de color vermell.

Un cop la llista esta ordenada se l'hi assigna la categoria de "blanc" a les "n" primeres peces de la llista, sent "n" el nombre de peces blanques que hi havia sobre el taulell al final del torn anterior. A la resta de peces se'ls hi assigna la categoria de "negres".

Finalment es mostra per pantalla la llista amb la informació i una imatge representativa

dels píxels consultats marcats en blanc sobre la foto del taulell.

FUNCIONS PER EXTRAPOLAR EL MOVIMENT REALITZAT

```
1. def taulell_inicial_CV():
2.     """
3.     Es crea el taulell inicial utilitzat per CV. Blanques = 2, Negres = 3.
4.     """
5.     t_ini = np.array([[3,3,3,3,3,3,3,3],
6.                        [3,3,3,3,3,3,3,3],
7.                        [0,0,0,0,0,0,0,0],
8.                        [0,0,0,0,0,0,0,0],
9.                        [0,0,0,0,0,0,0,0],
10.                       [0,0,0,0,0,0,0,0],
11.                       [2,2,2,2,2,2,2,2],
12.                       [2,2,2,2,2,2,2,2]], dtype=int)
13.
14.
15.     return t_ini
```

A partir de l'anàlisi d'imatge comentat anteriorment, el programa pot distingir on estan les peces i quin color tenen, però de moment no sap quin moviment ha fet l'humà.

Aquesta funció crea una matriu on es representen les posicions de cada color sobre el taulell. 0 = escac buit, 2 = peça blanca, 3 = peça negra.

Aquesta matriu s'utilitzarà en la funció que detectarà els moviments que s'explica a continuació.

```
1. def moviments_detectats_CV(t_ant, tau_CV):
2.     """
3.     Detecció de moviments
4.
5.     Com que els taulells t_ant i t_act guarden les posicions de les peces i el
6.     seu color en el format:
7.
8.     0 -> Buit
9.     2 -> Peça blanca
10.    3 -> Peça negra
11.
12.    Quan es resten els dos taulells (t_ant) menys (t_act) s'obté una matriu de
13.    moviments (mov), que pot contenir diversos valors cada un amb un significat dife
14.    rent:
15.
16.    0 -> No hi ha hagut cap moviment en aquest escac
17.
18.    3 -> Una peça negra ha sortit d'aquest escac i l'ha deixat buit
19.    2 -> Una peça blanca ha sortit d'aquest escac i l'ha deixat buit
20.
21.    1 -
22.    > Una peça blanca ha arribat a aquest escac i n'ha matat una de negra que hi er
23.    a avans
24.    -1 -
25.    > Una peça negra ha arribat a aquest escac i n'ha matat una de blanca que hi er
26.    a avans
27.
28.    -2 -> Una peça blanca ha arribat a aquest escac que avans estava buit
29.    -3 -> Una peça negra ha arribat a aquest escac que avans estava buit
30.
31.    Com es pot veure, els valors 3 i 2 indiquen les caselles de sortida i els v
32.    alors 1, -1, -2 i -3 indiquen caselles d'arribada.
33.
34.    Per tant, en qualsevol moviment, excepte l'enrocc, s'obtindrà una matriu (
35.    mov) que contindrà dos valors diferents a 0: un valor màxim (3 o 2) que
36.    correspondrà a la casella de sortida i un valor mínim (1, -1, -2 o -
37.    3) corresponent a la casella d'arribada.
38.
39.    """
40.
41.    M_mov = t_ant -
42.    tau_CV #Es busquen les diferències concretes entre el taulell anterior i l'actu
43.    al en forma de valors: (3,2,1,-1,-2,-3) segons quin hagi sigut el canvi.
44.
45.
46.    print('Matriu de moviments:')
47.    print(M_mov)
48.
49.
50.    c = np.count_nonzero(M_mov) #Conta quants valors "True" hi ha dins de la ma
51.    triu "dif", es a dir, la quantitat de diferències entre t_ant i t_act.
52.
53.    print('nº de diferències = ', c)
```

Aquesta funció detecta els moviments realitzats pel jugador humà a partir de les matrius de posicions de colors extreïdes de l'anàlisi d'imatge comentat anteriorment.

Per fer-ho es fa la següent operació: matriu de moviments = matriu de posicions anterior – matriu de posicions actual. Amb aquesta operació s'obté una matriu de les mateixes dimensions amb zeros en les posicions on no hi ha hagut cap canvi i valors entre 3 i -3 (exceptuant el zero) per a les posicions on hi hagin hagut canvis.

Seguidament es mostra per pantalla la matriu de moviments.

Es calculen el nombre de diferències trobades, que posteriorment s'utilitzaran per saber si el moviment és vàlid o es tracta d'una lectura errònia i si és vàlid, conèixer quin tipus de moviment és.

```
1.     if c == 2:
2.
3.         sortida_array = np.array(M_mov>1) # Matriu boleana on totes les posicio
ns son "False" escepte la de sortida que és "True"
4.         arribada_array = np.logical_and(M_mov!=0, M_mov<2) # Matriu boleana on
totes les posicions son "False" escepte la d'arribada que és "True"
5.
6.         sortida_index = np.argmax(sortida_array) # Es busca l'índex de la posic
ió "True" en la matriu de sortida
7.         arribada_index = np.argmax(arribada_array) # Es busca l'índex de la po
sició "True" en la matriu d'arribada
8.
9.         sortida_escac = chess.square_mirror(sortida_index)
10.        arribada_escac = chess.square_mirror(arribada_index)
11.
12.        #S'ha d'escriure un codi per detectar la promoció de peons, i que "prom
otion" s'iguali a una variable -> promotion = None/5 (5=reina)
13.
14.        mov = chess.Move(sortida_escac, arribada_escac, promotion=None, drop=No
ne) #es concatenen la posició inicial i la final per poder fer servir la funció
"board.push_san()"
15.
16.        if (arribada_index < 8) and (7 < sortida_index < 16):
17.
18.            mov_promoted = chess.Move(sortida_escac, arribada_escac, promotion=
5, drop=None)
19.
20.            if (mov_promoted in board.legal_moves) or (mov_promoted in board.ps
eudo_legal_moves):
21.                mov = mov_promoted
```

Si es detecten dues diferències en el taulell significa que hi ha hagut un moviment simple, s'ha eliminat una peça d'un escac i s'ha col·locat en un escac anteriorment buit o bé l'escac d'arribada estava ocupat per una peça de la màquina i s'ha eliminat i substituït per una del jugador.

Amb una sèrie d'operacions lògiques es detecten els escacs de sortida i d'arribada.

Aquestes posicions de sortida i arribada es transformen en els índexs que utilitza el taulell virtual i es transformen en un codi que pot ser llegit per aquest.

Seguidament es comprova si el moviment implica una promoció d'un peó, si fos el cas, s'afegiria la informació dins el codi del moviment.


```
1. elif c == 3:
2.
3.     sortida_array = np.array(M_mov == 2) # Matriu boleana on totes les posi
      cions son "False" escepte la de sortida que és "True" (només funciona per al to
      rn de les blanques)
4.     arribada_array = np.array(M_mov == -
      2) # Matriu boleana on totes les posicions son "False" escepte la d'arribada qu
      e és "True"
5.
6.     if (np.count_nonzero(sortida_array) == 1) and (np.count_nonzero(arribad
      a_array) == 1):
7.
8.         sortida_index = np.argmax(sortida_array) # Es busca l'índex de la p
      osició "True" en la matriu de sortida
9.         arribada_index = np.argmax(arribada_array) # Es busca l'índex de l
      a posició "True" en la matriu d'arribada
10.
11.        sortida_escac = chess.square_mirror(sortida_index)
12.        arribada_escac = chess.square_mirror(arribada_index)
13.
14.        # Falta la funció de promoció de peó
15.
16.        mov = chess.Move(sortida_escac, arribada_escac, promotion=None, dro
      p=None) #es concatenen la posició inicial i la final per poder fer servir la fu
      nció "board.push_san()"
17.
18.        if (mov in board.legal_moves) or (mov in board.pseudo_legal_moves):
19.
20.            if board.is_en_passant(mov): # Comprova que sigui un moviment d
      e captura al pas del peó
21.                pass
22.
23.            else:
24.                print ('Error de detecció en CV. S\'han detectat 3 diferènc
      ies, pero no coincideixen amb una captura al pas')
25.                mov = chess.Move.from_uci('0000')
26.            else:
27.                print ('Error de detecció en CV. S\'han detectat 3 diferències,
      pero no coincideixen amb una captura al pas')
28.                mov = chess.Move.from_uci('0000')
29.            else:
30.                print ('Error de detecció en CV. S\'han detectat 3 diferències, per
      o no coincideixen amb una captura al pas')
31.                mov = chess.Move.from_uci('0000')
```

En el cas que es detectin 3 diferències en les matrius de posicions, el moviment haurà estat una captura al pas del peó. Es comprova si és el cas i si és així es segueix el procediment per actualitzar el taulell. Sinó, es marca el moviment com a no vàlid.

Dispositiu automàtic per a jugar a escacs

Marçal Civill Cortada

```
1. elif c == 4: # L'enroc es l'unic moviment que permet moure dues peces en el ma
    teix torn
2.
3.     if np.argmax(M_mov) == 0: # argmax() retorna el rang del primer valor m
        àxim que troba
4.
5.         if bool(board.castling_rights & chess.BB_A1):
6.             mov = chess.Move.from_uci('e1c1') # El moviment d'enroc esta co
            dificat amb el moviment del rei a la casella corresponent
7.
8.         else:
9.             mov = chess.Move.from_uci('0000')
10.            print ('Ja no tens dret d'enroc amb A1')
11.
12.     elif np.argmax(M_mov) == 4:
13.
14.         if bool(board.castling_rights & chess.BB_H1):
15.             mov = chess.Move.from_uci('e1g1')
16.
17.         else:
18.             mov = chess.Move.from_uci('0000')
19.            print ('Ja no tens dret d'enroc amb H1')
20.
21.     else:
22.         print ('Error de detecció en CV. S\'ha detectat el moviment de dues
            peces a la vegada, pero no coincideixen amb les posicions d'un enroc')
23.         mov = chess.Move.from_uci('0000') # La representació UCI d'un movim
            ent nul es '0000'
```

Si es detecten 4 diferències significa que s'ha realitzat un enroc.

Es comprova que l'enroc sigui legal.

S no ho es marca el moviment com a no vàlid.

```
1. elif c == 0:
2.
3.     print ('ERROR de detecció en CV :|, no s\'ha detectat cap moviment')
4.     mov = chess.Move.from_uci('0000')
5.
6.
7. elif c == 1:
8.
9.     print('ERROR de detecció en CV :|, ha aparegut/desaparegut una peça mis
        teriosament')
10.    mov = chess.Move.from_uci('0000')
11.
12.
13. else:
14.     print('ERROR de detecció en CV :|, s\'han detectat més de 2 moviments
        la vegada')
15.     mov = chess.Move.from_uci('0000')
16.
17.
18. return mov
```

Per a qualsevol altre quantitat de diferències detectades, significa que o bé hi ha hagut un error en l'anàlisi d'imatge o bé el jugador ha realitzar un moviment il·legal. Per ambdós casos la jugada es marca com a il·legal i posteriorment es repetirà l'anàlisi o bé es demanarà al jugador que repeteixi la jugada.

FUNCIÓ DE CREACIÓ DEL MAPA NODAL

```
1. def espais(P_BN_llista, fila_pi, columna_pi, fila_pf, columna_pf):
2.
3.     """
4.     Ideal:
5.
6.     1r:
7.     Es crea la matriu de nodes per al "path finder".
8.     Es crea un node per cada centre d'escac, per cada vertex d'escac i per cada
        centre d'aresta d'escac.
9.     Per aquests nodes es calcularan els camins que seguiran les peces.
10.    Aquesta matriu s'omplirà inicialment de valors = 90. 90 es un valor de pixe
        ls superior a 42 que es el radi màxim detectble amb
11.    la funció "Hough Circles", actualment configurada a " maxRadius = 42 ").
12.    A efectes de càlcul, el valor 90 significa espai lliure per al pas sigui qu
        ina sigui la mida de la peça que s'estigui movent.
13.
14.    2n:
15.    El valor de tots els nodes centrals dels escacs ocupats, es canvia a 0, es
        a dir, hi ha 0 píxels de marge per passar (no es pot passar).
16.
17.    3r: Es calculs l'espai de pas per a les arestes dels escacs. Per fer-
        ho es resta el radi a la distancia entre el centre de la peça i l'aresta corres
        ponent.
18.    Després es compara aquest valor amb l'emmagatzemat anteriorment en aquest v
        ertex i s'escolleix el mínim. Es tria el mínim ja que una mateixa aresta esta
19.    compartida per 2 escacs, i el que limitarà el pas d'una peça per l'aresta, s
        erà la mínima distancia des de l'aresta a qualsevol de les peces del costat.
20.
21.    4t:
22.    El mateix que s'ha fet amb les arestes es fa amb els vertexs.
23.
24.    5è:
25.    Es crea una matriu boleana, on els valors d'espai superiors al radi de la p
        eça que es pot moure es convertiran en "True" i els iguals o inferiors a "False
        "
26.    A l'utilitzar el "path finder", els nodes amb valor de "True" es considerar
        an espais lliures i els que continguin valor "False" es consideraran "murs" o e
        spais bloquejats.
27.
28.    7è:
29.    No es fan els càlculs per a la peça d'origen, (la peça moguda) ja que no es
        pot bloquejar a ella mateixa.
30.
31.    8è:
32.    Quan es troba la peça d'origen es guarda el seu radi per posteriorment fer
        la comparació.
33.
34.
35.    Real:
36.
37.    Suposant que les peces no estan excessivament descantrades:
38.
39.    1r: Els nodes corresponents als centres i als vertexs d'escacs ocupats esta
        ran bloquejats. Valor = 1.
40.
41.    2n: Tots els vertexs estaran lliures. Valor = 0.
42.    """
```

Aquesta funció té per objectiu calcular la ruta mes curta possible per a moure una peça de la màquina de la posició inicial a la posició final. Aquesta informació es passarà posteriorment al programa que controla el moviment dels motors per moure la peça de forma automàtica.

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

```
1. espais = np.zeros((19,19), dtype=int)
2.
3.     for i in P_BN_llista:
4.         if (i['py'] == fila_pi and i['px'] == columna_pi) or (i['py'] == fila_p
f and i['px'] == columna_pf):
5.             """
6.             # Ideal:
7.             r_peça_moguda = i['r']
8.             """
9.             # Real (no es necessari comparar amb el radi de la peça)
10.            pass
```

Es crea una matriu que representa un mapa de nodes. Aquest mapa de nodes representa cada una xarxa de punts imaginària sobre el taulell per on es podran traçar els camins que seguirà la màquina. Aquests punts imaginàris es troben situats sobre cada vèrtex, centre d'aresta i entre del quadrat de cada escac del taulell i dels camins laterals de pas.

```
1. else:
2.     # Centre de l'escac
3.     """
4.     # Ideal:
5.     espais[2+2*(i['py'])][2+2*(i['px'])] = 0
6.     """
7.     # Real(es suposa que els escacs ocupats estan bloquejats i els espa
is bloquejats es marquen amb un 1, els lliures amb un 0.
8.     espais[2+2*(i['py'])][2+2*(i['px'])] = 1
```

Es marquen els nodes corresponents a centres d'escac bloquejats, és a dir, que no es podran utilitzar per passar.

```
1. # Arestes (començant per dalt i rotant en sentit horari)
2. """
3. # Ideal:
4. espais[1+2*(i['py'])][2+2*(i['px'])] = min(espais[1+2*(i['py'])][2+
2*(i['px'])], i['dt']-i['r'])
5. espais[2+2*(i['py'])][3+2*(i['px'])] = min(espais[2+2*(i['py'])][3+
2*(i['px'])], i['dr']-i['r'])
6. espais[3+2*(i['py'])][2+2*(i['px'])] = min(espais[3+2*(i['py'])][2+
2*(i['px'])], i['db']-i['r'])
7. espais[2+2*(i['py'])][1+2*(i['px'])] = min(espais[2+2*(i['py'])][1+
2*(i['px'])], i['dl']-i['r'])
8. """
9. # Real (es suposarà que totes les arestes d'escacs ocupats estaran
bloquejats):
10. espais[1+2*(i['py'])][2+2*(i['px'])] = 1
11. espais[2+2*(i['py'])][3+2*(i['px'])] = 1
12. espais[3+2*(i['py'])][2+2*(i['px'])] = 1
13. espais[2+2*(i['py'])][1+2*(i['px'])] = 1
```

Es fa el mateix per als centres d'aresta.

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

```
1. # Vertexs (començant per dalt-esquerra i rotant en sentit horari)
2. """
3.     # Ideal:
4.     espais[1+2*(i['py'])][1+2*(i['px'])] = min(espais[1+2*(i['py'])][1+
5. 2*(i['px'])], (math.sqrt((i['dt'])**2+(i['dl'])**2))-i['r'])
6.     espais[1+2*(i['py'])][3+2*(i['px'])] = min(espais[1+2*(i['py'])][3+
7. 2*(i['px'])], (math.sqrt((i['dt'])**2+(i['dr'])**2))-i['r'])
8.     espais[3+2*(i['py'])][3+2*(i['px'])] = min(espais[3+2*(i['py'])][3+
9. 2*(i['px'])], (math.sqrt((i['db'])**2+(i['dr'])**2))-i['r'])
10.    espais[3+2*(i['py'])][1+2*(i['px'])] = min(espais[3+2*(i['py'])][1+
11. 2*(i['px'])], (math.sqrt((i['db'])**2+(i['dl'])**2))-i['r'])
12.    """
13.    # Real (es suposarà que tots els vertexs estan lliures
14.    espais[1+2*(i['py'])][1+2*(i['px'])] = 0
15.    espais[1+2*(i['py'])][3+2*(i['px'])] = 0
16.    espais[3+2*(i['py'])][3+2*(i['px'])] = 0
17.    espais[3+2*(i['py'])][1+2*(i['px'])] = 0
```

I el mateix per als vèrtexs.

```
1. """
2.     # Ideal:
3.     espais = espais > r_peça_moguda # Creació de la matriu boleana d'"espais i
4.     murs"
5.     """
6.     # Real (es pot fer això o assignar True/False en comptes de 0/1 en els apar
7.     tats anteriors, però 0/1 va millor per visualitzar la matriu)
8.     #espais = espais == 0
9.
10.    return espais
```

Finalment es retorna la matriu d'espais per ser utilitzada per el buscador de camins.

FUNCIÓ PER LOCALITZAR LA POSICIÓ DE L'ACTUADOR

Inicialment es va escriure el programa per ser utilitzat conjuntament entre la Raspberry Pi i un Arduino. La funció dins de la Raspberry era la següent:

```
#!/usr/bin/env python

import serial
import time

ser = serial.Serial('/dev/ttyACM0', 9600, timeout=1)
# time.sleep(2) # La comunicació serie triga uns 2 segons a
establir-se.
# En principi amb el temps que triga el programa principal a
utilitzar aquesta funció, no serà necessari el .sleep()

def xyArduino():

    ser.write(bytearray(b'A')) # Envia un Byte a l'Arduino per
demanar info (en aquest cas una "A")

    x_y = ser.readline() # Es llegeix una línia sencera (tots els
Byts fins que es troba un final de línia ('\n'))

    x_y = x_y.strip()      # S'eliminen els caràcters de canvi de línia
    x_y = x_y.decode()     # Es descodifica desde Bytes
    x_y = x_y.split(",")   # Es separen els elements per comes i es
guarden en una llista

    x,y = x_y

    return int(x),int(y)

while True:
    a = int(input('Escriu "0" per no llegir o "1" per llegir: '))
    print('in waiting 0: ', ser.in_waiting)

    if a == 1:
        x,y = xyArduino()
        print (x)
        print (y)
```

Dispositiu automàtic per a jugar a escacs

Marçal Civill Cortada

I la funció dins l'Arduino aquesta altre:

```
// Les equacions del moviment es poden trobar a la web:
http://corexy.com/theory.html
// Totes les referències (A, B, increments i coordenades) estan
representades en l'esquema final d'aquesta web

// Declaració de variables globals constants
int x_inicial = 0.0;
int y_inicial = 0.0;

// Dimensions del taulell en mm.
int xyMin[2] = {0.0,0.0}; // Coordenada (x,y)
int xyMax[2] = {450.0,510.0}; // Coordenada (x,y)

// Ajustar el rang fent proves amb els potenciòmetres quan tot
estigui muntat desde la lectura en la posició mínima a la màxima
// Dependrà del perímetre de la politja i de la llargada/amplada
del taulell, d'aquí s'otindran les voltes màximes i finament el
valor de lectura màxim
float pot_A_rang[2] = {0.0,1023.0};
float pot_B_rang[2] = {0.0,1023.0};

// Longituds de corretja inicials
int L_inicial_B = xyMax[0];
int L_inicial_A = 0;

// Byte d'entrada (per a la comunicació serie)
int inByte = 0; // incoming serial byte

void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native USB port
    only
  }
}
```

```
void loop() {  
  // if we get a valid byte, read analog ins:  
  if (Serial.available() > 0) {  
  
    // get incoming byte:  
    inByte = Serial.read(); // S'elimina el Bite d'entrada perquè el  
    bucle no continui  
  
    // read the input on analog pins 0 and 1:  
    int pot_B = analogRead(A0);  
    int pot_A = analogRead(A1);  
  
    // Convertir les dues lectures analògiques dels potenciòmetres a  
    l'increment de corretja A i B respectivament (en mm).  
  
    float L_actual_B = pot_B * ( (xyMax[0]+xyMax[1]) /  
    (pot_B_rang[1]- pot_B_rang[0]) );  
    float L_actual_A = pot_A * ( (xyMax[0]+xyMax[1]) /  
    (pot_A_rang[1]- pot_A_rang[0]) );  
  
    float increment_A = L_actual_A - L_inicial_A;  
    float increment_B = (L_actual_B - L_inicial_B)*(-1);  
  
    float increment_x = 0.5 * (increment_A + increment_B);  
    float increment_y = 0.5 * (increment_A - increment_B);  
  
    int x = x_inicial + increment_x;  
    int y = y_inicial + increment_y;  
  
    Serial.print(x);    // S'utilitza .print en comptes de .write  
    perquè la primera envia en format text i la segon en format enter  
    Serial.print(','); // Separador. (posteriorment els dos valors  
    es separaran amb (.split(',') [python3]))  
    Serial.println(y); // S'utilitza .println en comptes de print  
    perquè la primera afegeix un final de línia i la segona no  
  }  
}
```

En aquest cas l'arduino s'encarregava de llegir els potenciòmetres, calcular les coordenades i enviar-les a la Raspberry.

Més endavant es va decidir utilitzar un PIC12F1822 en comptes d'un Arduino per reduir electrònica, però el codi encara s'ha d'adaptar al nou format.

El programa introduït en el PIC consisteix en:

1. Esperar que la Raspberry Pi envii un bit a través del port sèrie.
2. Quan el PIC rep un bit, aquest llegirà els dos potenciòmetres i convertirà les lectures compreses entre 0 i 5V a dos valors entre 0 i 1023 respectivament.
3. Un cop completat aquest procés enviarà els dos valors a la Raspberry Pi mitjançant el port sèrie.

Aquí apareix una limitació. A través del port sèrie es pot enviar un Byte cada vegada i cada Byte contindrà 8 bits a causa de l'arquitectura del PIC. Un número de 8 bits en codi binari pot representar un valor entre 0 i 255 en codi decimal (entre 00000000 i 11111111 en codi binari). Per arribar al valor de 1023 (1111111111 en binari) es requereixen 10 bits, per tant el PIC haurà d'enviar dos Bytes consecutius a la Raspberry que acabaran agrupant 16 bits en total. Aquests 16 bits consecutius poden representar un valor entre 0 i 65535 suficient per representar un valor comprès entre 0 i 1023.

4. Un cop la Raspberry hagi rebut 4 Bytes (2 per cada potenciòmetre) aquests hauran quedat emmagatzemats al port d'entrada i els començarà a llegir per ordre d'arribada.

Al port sèrie de la Raspberry Pi arribaran els 4 Bytes de 8 bits cada un en l'ordre següent, de dalt a baix:

$d_{15}d_{14}d_{13}d_{12}d_{11}d_{10}d_{09}d_{08}$

$d_{07}d_{06}d_{05}d_{04}d_{03}d_{02}d_{01}d_{00}$

$d_{15}d_{14}d_{13}d_{12}d_{11}d_{10}d_{09}d_{08}$

$d_{07}d_{06}d_{05}d_{04}d_{03}d_{02}d_{01}d_{00}$

La primera parella de nombres correspon al valor del primer potenciòmetre i el segon parell de nombres al segon.

Els dos últims bits del primer Byte de la llista corresponen al primer i segon dígit del total de 10 dígits que componen el número 1023 en codi binari. Per convertir el número en format binari binari a format decimal s'ha de multiplicar cada dígit per 2 elevat al índex corresponent a la posició de cada dígit i sumar tots els resultats:

$$n^{\circ} \text{ decimal} = d_0 \times 2^0 + d_1 \times 2^1 + d_2 \times 2^2 + \dots + d_9 \times 2^9$$

Només cal calcular fins a l'índex 9 ja que el valor màxim que s'enviarà és el 1023 que en binari ocupa 10 dígits, per tant els dígits del 10 al 15 sempre seran 0.

FUNCIO BUSCADOR DE CAMINS



(En elaboració)

FUNCIO DE MOVIMENT DELS MOTORS PAS A PAS



(En elaboració)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

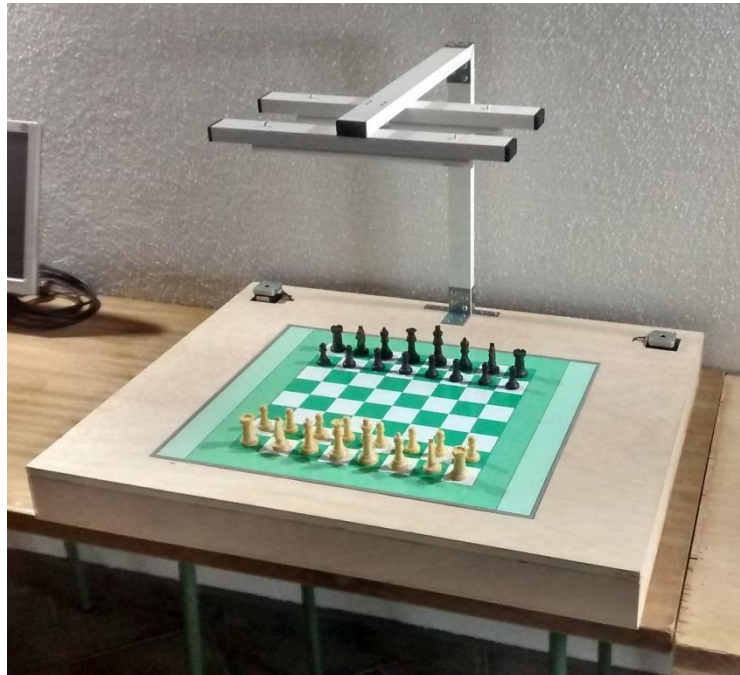
FUNCIÓ DE MOVIMENT DEL SERVOMOTOR



(En elaboració)

Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada

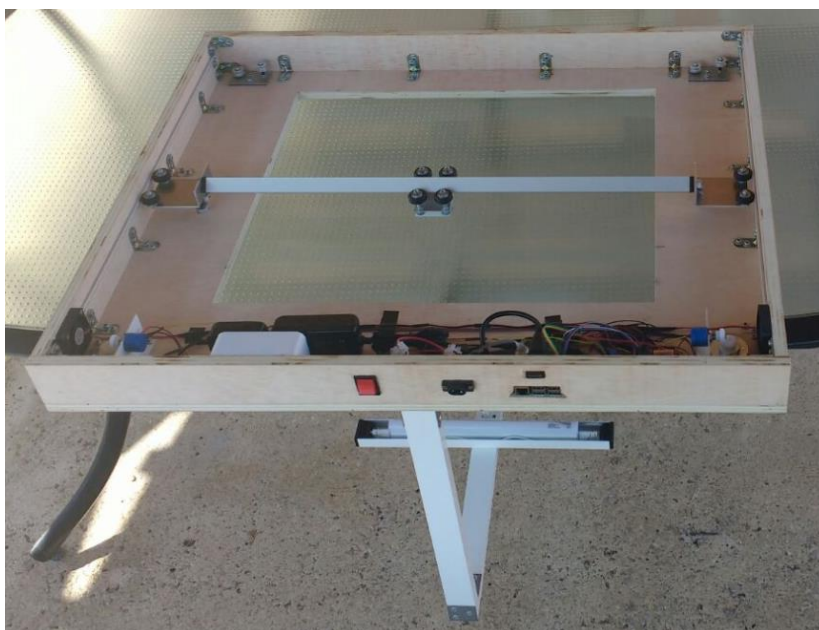
PROTOTIP DESENVOLUPAT. RECULL D'IMATGES REPRESENTATIVES.



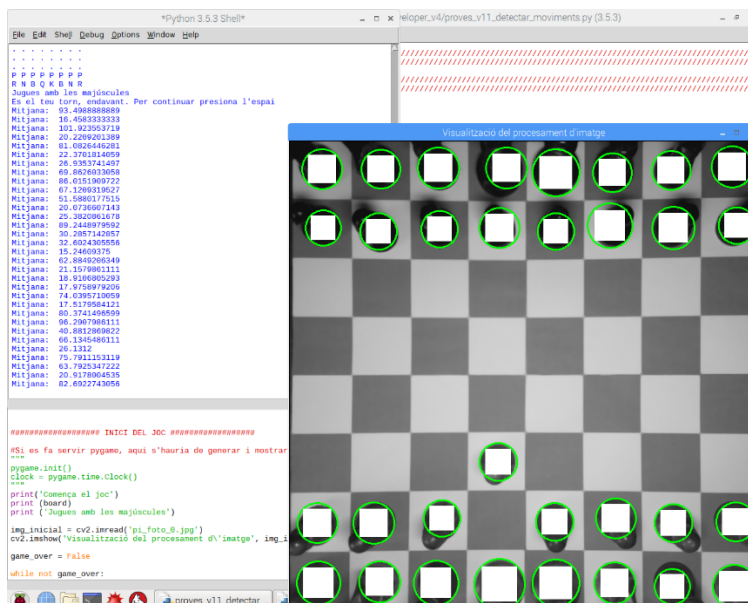
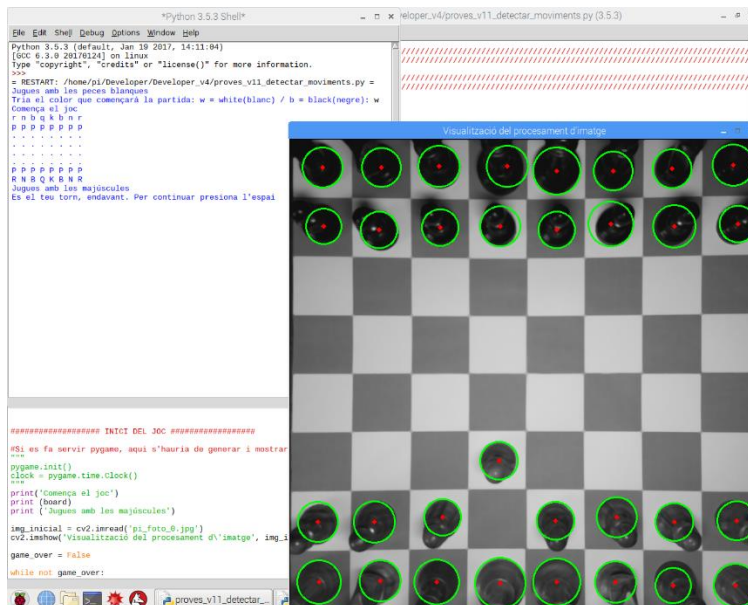
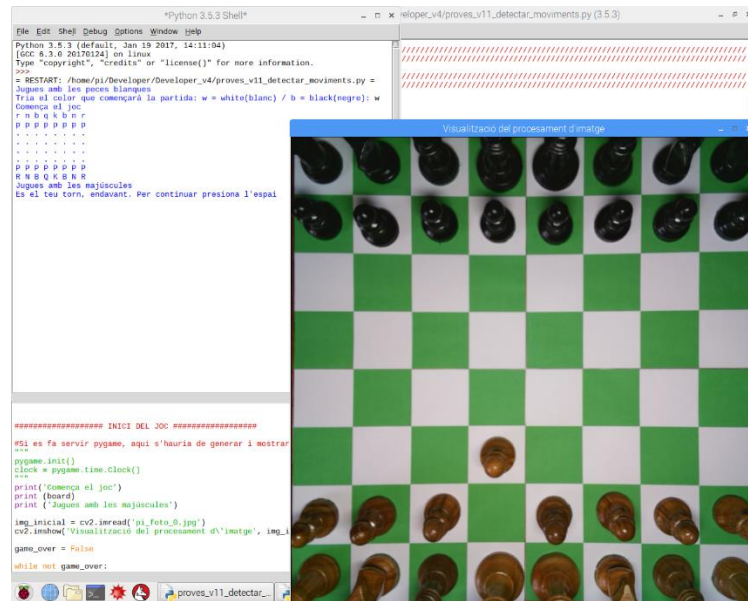
Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada



Dispositiu automàtic per a jugar a escacs
Marçal Civill Cortada



Dispositiu automàtic per a jugar a escacs Marçal Civill Cortada



BIBLIOGRAFIA

- [1] Arduino & Raspberry Pi chess computer using Stockfish & reed switches. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://chess.forththerapy.co.uk/home/>.
- [2] Larregay et al. - 2016 - Robot ajedrecista sistema mecatrónico aplicado a la toma de decisiones.pdf [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: http://sedici.unlp.edu.ar/bitstream/handle/10915/58393/Documento_completo.pdf-PDFA.pdf?sequence=1.
- [3] Matuszek et al. - Gambit A Robust Chess-Playing Robotic System.pdf [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://rse-lab.cs.washington.edu/postscripts/chess-icra-11.pdf>.
- [4] MEYER, J., [sense data]. Raspberry Turk. [en línia]. [Consultat: 25 maig 2018]. Disponible a: <http://www.raspberryturk.com>.
- [5] SAM DONALD, [sense data]. Stonewall - Chess Computer Vision [en línia]. [Consultat: 25 maig 2018]. Disponible a: https://www.youtube.com/watch?v=uDw_vCNbx-l.
- [6] Square Off - Connecting the world on a telerobotic chessboard. Square Off [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://squareoffnow.com>.
- [7] Square Off - World's Smartest Chess Board. Kickstarter [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.kickstarter.com/projects/infivention/square-off-worlds-smartest-chess-board-relaunched>.
- [8] Unger - Magnus the Chess Robot.pdf [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://daim.idi.ntnu.no/masteroppgaver/010/10695/masteroppgave.pdf>.
- [9] Urting - MarineBlue A Low-Cost Chess Robot.pdf [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://pdfs.semanticscholar.org/656d/64d345d252208a40cb1068280d16ed227619.pdf>.
- [10] Built-in Types — Python 3.6.5 documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs.python.org/3/library/stdtypes.html#binary-sequence-types-bytes-bytearray-memoryview>.
- [11] Arduino - SerialCallResponse. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.arduino.cc/en/Tutorial/SerialCallResponse>.
- [12] Arduino - SerialCallResponseASCII. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.arduino.cc/en/Tutorial/SerialCallResponseASCII>.
- [13] Arduino Reference. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.arduino.cc/reference/en/language/functions/communication/serial/>.

- [14] Detección de líneas y círculos usando la transformada de Hough con OpenCV. *Detección de líneas y círculos usando la transformada de Hough con OpenCV* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://acodigo.blogspot.com.es/2017/09/deteccion-de-lineas-y-circulos-usando.html>.
- [15] Funciones de dibujo OpenCV Python. *Funciones de dibujo OpenCV Python* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://acodigo.blogspot.com.es/2017/06/funciones-de-dibujo-opencv-python.html>.
- [16] Getting started with picamera - Introduction | Raspberry Pi Projects. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://projects.raspberrypi.org/en/projects/getting-started-with-picamera>.
- [17] GPIO - Raspberry Pi Documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/documentation/usage/gpio/README.md>.
- [18] GPIO in Python - Raspberry Pi Documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/documentation/usage/gpio/python/README.md>.
- [19] gpiozero — Gpiozero 1.4.1 Documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://gpiozero.readthedocs.io/en/stable/>.
- [20] Instalar OpenCV en Python. *Instalar OpenCV en Python* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://acodigo.blogspot.com.es/2017/06/instalar-opencv-en-python.html>.
- [21] Instalar OpenCV en Raspberry Pi – robologs. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://robologs.net/2014/04/25/instalar-opencv-en-raspberry-pi-2/>.
- [22] Install RPi.GPIO Python Library. *Raspberry Pi Spy* [en línia], 2012. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi-spy.co.uk/2012/05/install-rpi-gpio-python-library/>.
- [23] Introducción OpenCV Python. *Introducción OpenCV Python* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://acodigo.blogspot.com.es/2017/06/introduccion-opencv-python.html>.
- [24] Introduction - A Byte of Python. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://python.swaroopch.com/>.
- [25] Manipular pixeles OpenCV Python. *Manipular pixeles OpenCV Python* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://acodigo.blogspot.com.es/2017/06/manipular-pixeles-opencv-python.html>.
- [26] MONK, S., [sense data]. Raspberry Pi and Arduino. [en línia]. [Consultat: 25 maig 2018]. Disponible a: <http://www.doctormonk.com/2012/04/raspberry-pi-and-arduino.html>.
- [27] NumPy basics — NumPy v1.14 Manual. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs.scipy.org/doc/numpy/user/basics.html>.

- [28] NumPy Reference — NumPy v1.14 Manual. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs.scipy.org/doc/numpy/reference/>.
- [29] Numpy_Example_List - SciPy wiki dump. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: http://scipy.github.io/old-wiki/pages/Numpy_Example_List.
- [30] OpenCV: Basic Operations on Images. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://docs.opencv.org/3.3.0/d3/df2/tutorial_py_basic_ops.html.
- [31] OpenCV: Camera Calibration. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://docs.opencv.org/3.3.0/dc/dbb/tutorial_py_calibration.html.
- [32] OpenCV: Changing Colorspaces. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://docs.opencv.org/3.3.0/df/d9d/tutorial_py_colorspaces.html.
- [33] OpenCV documentation index. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs.opencv.org/>.
- [34] OpenCV: Drawing Functions in OpenCV. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://docs.opencv.org/3.3.0/dc/da5/tutorial_py_drawing_functions.html.
- [35] OpenCV: Geometric Transformations of Images. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://docs.opencv.org/3.3.0/da/d6e/tutorial_py_geometric_transformations.html.
- [36] OpenCV: Getting Started with Images. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://docs.opencv.org/3.3.0/dc/d2e/tutorial_py_image_display.html.
- [37] OpenCV: Hough Circle Transform. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://docs.opencv.org/3.3.0/da/d53/tutorial_py_houghcircles.html.
- [38] OpenCV: Install OpenCV-Python in Windows. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://docs.opencv.org/3.3.0/d5/de5/tutorial_py_setup_in_windows.html.
- [39] OpenCV: Introduction to OpenCV-Python Tutorials. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://docs.opencv.org/3.3.0/d0/de3/tutorial_py_intro.html.
- [40] OpenCV: Smoothing Images. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://docs.opencv.org/3.3.0/d4/d13/tutorial_py_filtering.html.
- [41] OpenCV-Python Tutorials — OpenCV-Python Tutorials 1 documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_tutorials.html.

- [42] PEP 8 -- Style Guide for Python Code. *Python.org* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.python.org/dev/peps/pep-0008/>.
- [43] picamera — Picamera 1.13 Documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://picamera.readthedocs.io/en/release-1.13/>.
- [44] pyserial. *PyPI* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://pypi.org/project/pyserial/>.
- [45] Python - Raspberry Pi Documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/documentation/usage/python/>.
- [46] Python Serial: How to use the read or readline function to read more than 1 character at a time. *Stack Overflow* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://stackoverflow.com/questions/16077912/python-serial-how-to-use-the-read-or-readline-function-to-read-more-than-1-char>.
- [47] Quickstart tutorial — NumPy v1.14 Manual. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs.scipy.org/doc/numpy/user/quickstart.html>.
- [48] Raspberry Pi Pinout Diagram: Navigating the Raspberry Pi 3 Model B. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.jameco.com/Jameco/workshop/circuitnotes/raspberry-pi-circuit-note.html>.
- [49] raspberry-gpio-python / Wiki / Home. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://sourceforge.net/p/raspberry-gpio-python/wiki/Home/>.
- [50] Simple Guide to the Raspberry Pi GPIO Header and Pins. *Raspberry Pi Spy* [en línia], 2012. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi-spy.co.uk/2012/06/simple-guide-to-the-rpi-gpio-header-and-pins/>.
- [51] Structured arrays — NumPy v1.14 Manual. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs.scipy.org/doc/numpy/user/basics.rec.html>.
- [52] Subclassing ndarray — NumPy v1.14 Manual. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs.scipy.org/doc/numpy/user/basics.subclassing.html>.
- [53] Tabla de Contenidos — Tutorial de Python 3.6.3 documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://docs.python.org.ar/tutorial/3/index.html>.
- [54] The Python Standard Library — Python 3.6.5 documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs.python.org/3/library/>.
- [55] The Python Tutorial — Python 3.6.5 documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs.python.org/3/tutorial/>.
- [56] Transformaciones Geométricas. *Transformaciones Geométricas* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://acodigo.blogspot.com.es/2017/07/transformaciones-geometricas.html>.

- [57] TUTORIALSPPOINT.COM, [sense data]. Python Tutorial. *www.tutorialspoint.com* [en línia]. [Consultat: 25 maig 2018]. Disponible a: <https://www.tutorialspoint.com/python/index.htm>.
- [58] Welcome to opencv documentation! — OpenCV 2.4.13.6 documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs.opencv.org/2.4.13.6/>.
- [59] Welcome to pySerial's documentation — pySerial 3.0 documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://pythonhosted.org/pyserial/>.
- [60] chessprogramming - home. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://chessprogramming.wikispaces.com/>.
- [61] chessprogramming - Stockfish. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://chessprogramming.wikispaces.com/Stockfish>.
- [62] Home - Stockfish - Open Source Chess Engine. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://stockfishchess.org/>.
- [63] *Stockfish: UCI chess engine* [en línia], 2018. C++. S.l.: official-stockfish. [Consultat: 25 maig 2018]. Disponible a: <https://github.com/official-stockfish/Stockfish>.
- [64] python-chess: a pure Python chess library — python-chess 0.23.5 documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://python-chess.readthedocs.io/en/latest/>.
- [65] Getting started with the Raspberry Pi - Introduction | Raspberry Pi Projects. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://projects.raspberrypi.org/en/projects/raspberry-pi-getting-started>.
- [66] Help Guides and Resources - How to Use Raspberry Pi. *Raspberry Pi* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/help/>.
- [67] Raspberry Pi 3 Model B. *Raspberry Pi* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>.
- [68] Raspberry Pi Downloads - Software for the Raspberry Pi. *Raspberry Pi* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/downloads/>.
- [69] Raspberry Pi Hardware Guide | Raspberry Pi Learning Resources. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/learning/hardware-guide/>.
- [70] Raspberry Pi Software Guide | Raspberry Pi Learning Resources. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/learning/software-guide/>.
- [71] Raspbian - Raspberry Pi Documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/documentation/raspbian/>.

- [72] Camera Module - Raspberry Pi Documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/documentation/hardware/camera/README.md>.
- [73] Power Supply - Raspberry Pi Documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md>.
- [74] SD cards - Raspberry Pi Documentation. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.raspberrypi.org/documentation/installation/sd-cards.md>.
- [75] 2.2. Estructura de los Robots: Robot de Configuración Cartesiana (PPP). [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.udesantiagoovirtual.cl/moodle2/mod/book/view.php?id=24911>.
- [76] 3590.pdf [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.bourns.com/pdfs/3590.pdf>.
- [77] 0900766b8157a694.pdf [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://docs-emea.rs-online.com/webdocs/157a/0900766b8157a694.pdf>.
- [78] 40001413E.pdf [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://ww1.microchip.com/downloads/en/DeviceDoc/40001413E.pdf>.
- [79] CoreXY | Cartesian Motion Platform. [en línia], [sense data]. [Consultat: 25 maig 2018 a]. Disponible a: <http://corexy.com/index.html>.
- [80] CoreXY | Cartesian Motion Platform. [en línia], [sense data]. [Consultat: 25 maig 2018 b]. Disponible a: <http://corexy.com/theory.html>.
- [81] Delrin V Wheel. *OpenBuilds Part Store* [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://openbuildspartstore.com/delrin-v-wheel/>.
- [82] drv8825.pdf [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <https://www.pololu.com/file/0J590/drv8825.pdf>.
- [83] HB_Stepper_Motor_E.pdf [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: http://www.motionking.com/download/HB_Stepper_Motor_E.pdf.
- [84] RepRap - RepRapWiki. [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://reprap.org/>.
- [85] SG90.pdf [en línia], [sense data]. [Consultat: 25 maig 2018]. Disponible a: <http://akizukidenshi.com/download/ds/towerpro/SG90.pdf>.